

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE  
LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE



Laboratoire Commande des processus



Département du Génie Electrique  
Spécialité Automatique

**Projet de fin d'études**

Pour l'obtention du diplôme  
**d'Ingénieur d'Etat en Automatique**

Intitulé

**Synthèse et implémentation pratique de lois de  
commande hybrides multi-contrôleurs.  
Application au simulateur d'hélicoptère TRMS**

Etudié par :

**Mr ABDESLEM Hacem**

**Mlle BENALLEGUE Leila**

Proposé et dirigé par :

**Pr D. BOUKHETALA**

**Pr F. BOUDJEMA**

**Juin 2008**

Ecole Nationale Polytechnique 10, avenue Hassen Badi BP182 El-Harrach  
16200 Alger (Algérie)

## الملخص

هذا العمل يتطرق إلى مشكل التحكم في مقلد الطيران لطائرة عمودية عن طريق أنظمة التحكم المهجنة و المتعددة التحكمات. أولاً تم وصف أجزاء الجهاز، ثم اقتراح النموذج الرياضي لهذا الأخير. بعد ذلك، عدة أنظمة تحكم محلية (تحكم  $H_{\infty}$ ، تحكم RST، تحكم رجوع الحالة) تم حسابها باستعمال النماذج المحلية الخطية المستخرجة من النموذج الاخطي للجهاز لاختيار التحكم المناسب و الذي يطبق للجهاز، ثم عرض نوعين من قوانين التبديل: التبديل واحد من الكل و التبديل الغامض و هذا الاختيار يتم حسب بيئة الجهاز. و في النهاية تم عرض النتائج التجريبية المتحصل عليها عن تطبيق أنظمة التحكم التي تم حسابها سابقاً على الجهاز الحقيقي

**كلمات مفتاح:** مقلد الطيران لطائرة عمودية، التحكم المهجن المتعدد التحكمات، منطق التبديل، التحكم  $H_{\infty}$ ، التحكم RST، التبديل واحد من الكل، التبديل الغامض، النماذج المحلية.

## Résumé

Ce projet de fin d'études porte sur la commande hybride multi-contrôleurs du simulateur d'hélicoptère. Après un bref aperçu sur ses constituants, un modèle mathématique est proposé. Ainsi, trois types de commandes locales ( $H_{\infty}$  et RST et par retour d'états) sont calculées sur la base de modèles locaux obtenus par linéarisation des équations de la dynamique du système. Deux types de commutations sont alors proposés pour sélectionner un ou plusieurs contrôleurs locaux dans le but de constituer le signal de commande appliqué au système. Des simulations sur le modèle du système sont effectuées, puis une implémentation sur le simulateur est effectuée afin d'appuyer les résultats obtenus. Enfin, une étude comparative entre les différentes approches et différentes commandes est proposée.

**Mots clés :** simulateur d'hélicoptère, commande hybride multi contrôleurs, logique de commutation, commande  $H_{\infty}$ , commande RST, commutation tout ou rien, commutation floue, grafcet, modèles locaux.

## Abstract

This work relates to the hybrid multi-controller control of a helicopter simulator (TRMS). After a short description of the system, a mathematical model of this one is proposed. Then, three kinds of local controls ( $H_{\infty}$ , RST and state feed back) will be synthesized by using the local models resulting from the linearization of the process's dynamic model. The multi-controller control allows the switching of control laws according to the environment of the process. So, two types of switching logic are given for the selection of one or many local controllers in order to make the control signal applied to the system. A many simulations on the model of the system are done, then real time experimentation on the Twin Rotor MIMO System is presented and some experimental results are shown. Finally, a comparative study between the different approaches of controls is done.

**Key words:** Helicopter simulator, Hybrid multi-controller control, switching logic,  $H_{\infty}$  control, RST control, all or nothing switching, fuzzy switching, grafcet, local models.

# Remerciements

*Ce travail a été effectué au sein du Laboratoire de Commande des Processus au département du Génie Electrique de l'Ecole Nationale Polytechnique d'Alger sous la direction de nos deux promoteurs Mr D. BOUKHETALA et Mr F. BOUDJEMA Aux quels nous tenons à exprimer nos plus sincères remerciements pour leurs précieux conseils, leur aide et leur patience tout au long de notre travail*

*Nous tenons à remercier Mr TADJINE pour ses conseils et ses orientations, Mr BENMENSOUR pour son intérêt, sa disponibilité et sa bienveillance sans oublier Mr MAHMOUDI, qui n'a pas hésité à sacrifier de son temps pour nous apporter son aide.*

*Merci à Mr ILLOUL pour l'intérêt qu'il a manifesté à notre travail. Nous tenons également à remercier Mr BOUCHERIT pour ses encouragements.*

*Nous exprimons notre plus sincère gratitude à Mr. BENALLEGUE Abdelaziz, pour son aide, sa disponibilité et ses conseils lors de l'élaboration de ce mémoire.*

*Nous remercions les membres du jury qui nous ont fait l'honneur d'examiner notre travail.*

*Nous remercions nos enseignants de l'Ecole Nationale Polytechnique pour leurs efforts voués à nous transmettre le savoir, en particulier ceux du département Génie Electrique et Electronique.*

*Nous tenons à exprimer notre très grande gratitude, et notre profonde affection à nos chers parents pour leurs encouragements, leur patience et leur grand soutien, durant toutes ces années d'études.*

*Nous remercions également tous nos amis et camarades, pour leurs encouragements et leur précieux soutien.*

# *Dédicaces*

*À mes chers parents,*

*À mes tendres frères,*

*À ma chère cousine Yağout,*

*À mes fidèles amies Teldja et Lamia,*

*À tous mes proches,*

*À tous ceux qui sont chers à mon cœur.*

*À ceux que j'aime et qui m'aiment*

*Affectueusement*

*Leila*

# *Dédicaces*

*À mes parents*

*À mes grands-parents*

*À mes sœurs*

*À mes tantes*

*À mes oncles*

*À tous ceux que j'aime et qui m'aiment*

*Hacen*

# Sommaire général

## Introduction générale

### Chapitre I : Présentation du simulateur d'hélicoptère

<b>I.1 Introduction</b> .....	3
<b>I.2 Principe de vol d'un hélicoptère</b> .....	3
<b>I.3 Description du simulateur</b> .....	5
I.3.1 Propulseurs.....	7
I.3.2. Commande des moteurs.....	7
I.3.3 Les encodeurs optiques .....	8
I.3.4 La base du TRMS .....	9
I.3.5 Boîtier marche/arrêt .....	9
I.3.6 La SCSI BOX .....	10
I.3.7 Carte d'acquisition : ADVANTECH PCI 1711 .....	10
<b>I.4 Environnement de développement</b> .....	11
<b>I.5 Conclusion</b> .....	15

### Chapitre II: Modélisation du TRMS

<b>II.1 Introduction</b> .....	16
<b>II.2 Modèle non linéaire</b> .....	17
II.2.1 Modélisation du sous système d'élévation .....	17
II.2.1.1 Moment de la gravitation $M_{v1}$ .....	18
II.2.1.2 Moment de la force aérodynamique $M_{v2}$ .....	20
II.2.1.3 Moment de la force centrifuge $M_{v3}$ .....	20
II.2.1.4 Moment de la force de friction $M_{v4}$ .....	21
II.2.1.5 Moment d'inertie $J_v$ .....	21

II.2.2	Modélisation du sous système d'azimut.....	22
II.2.2.1	Influence de l'angle d'élévation sur le moment d'inertie.....	22
II.2.2.2	Influence de l'angle d'élévation sur le bras de levier des forces.....	23
II.2.2.3	Moment de la force aérodynamique $M_{h1}$ .....	24
II.2.2.4	Moment de la force de friction $M_{h2}$ .....	24
II.2.2.5	Moment d'inertie $J_h$ .....	25
II.2.3	Modélisation des propulseurs.....	25
II.2.4	Effet gyroscopique.....	27
<b>II.3</b>	<b>Modèle d'état non linéaire</b> .....	<b>28</b>
<b>II.4</b>	<b>Le modèle découplé</b> .....	<b>30</b>
II.4.1	Modèle 1 DDL vertical.....	31
II.4.2	Modèle 1 DDL horizontal.....	31
<b>II.5</b>	<b>Simulations</b> .....	<b>32</b>
<b>II.6</b>	<b>Conclusion</b> .....	<b>35</b>

## Chapitre III: Quelques aspects de la commande multi contrôleurs

<b>III.1</b>	<b>Introduction</b> .....	<b>36</b>
<b>III.2</b>	<b>Concept général de la commande multi contrôleurs</b> .....	<b>36</b>
<b>III.3</b>	<b>Commande multi contrôleurs et multi modèle</b> .....	<b>38</b>
<b>III.4</b>	<b>Commutation «franche»</b> .....	<b>38</b>
III.4.1	approche directe.....	38
III.4.1	approche indirecte.....	39
<b>III.5</b>	<b>Gestion des Commutation</b> .....	<b>40</b>
III.5.1	Réinitialisation de la commande.....	40
III.5.2	Poursuite de la commande.....	43
<b>III.6</b>	<b>Commutation «floue» et approche directe</b> .....	<b>45</b>
<b>III.7</b>	<b>Modèle Flou TS</b> .....	<b>46</b>
<b>III.8</b>	<b>Construction d'un modèle flou TS</b> .....	<b>48</b>
<b>III.7</b>	<b>Conclusion</b> .....	<b>48</b>

## Chapitre IV: Rappel Théorique sur les commandes utilisées

<b>IV.1 Introduction</b> .....	49
<b>IV.2 La commande <math>H^\infty</math></b> .....	49
IV.2.1 La norme $H^\infty$ .....	49
IV.2.2 Analyse des principaux transferts d'un asservissement.....	50
IV.2.3 Le problème $H^\infty$ standard .....	52
IV.2.4 Mise en œuvre par le choix de pondérations fréquentielles.....	53
<b>IV.3 La commande RST</b> .....	54
IV.3.1 Approche décentralisé .....	55
IV.3.1.1 modèle du système .....	55
IV.3.1.2 Structure du régulateur .....	55
IV.3.1.3 placement de pôle .....	57
IV.3.1.4 filtrage de la consigne .....	58
IV.3.2 Approche centralisé.....	59
<b>IV.4 Commande par retour d'état</b> .....	60
<b>IV.5 La logique floue</b> .....	62
IV.5.1 Principe et définitions .....	62
IV.5.2 Ensembles flous .....	64
IV.5.3 Variables linguistiques.....	64
IV.5.4 Fonction d'appartenance .....	64
IV.5.5 Raisonnement flou .....	65
IV.5.6 Commande par la logique floue .....	66
IV.5.7 Contrôleur flou.....	66
IV.5.8 Types de régulateurs flous .....	68
IV.5.8.1 Régulateur de type Mamdani.....	68
IV.5.8.2 Régulateur de type Sugeno .....	68
<b>IV.6 Conclusion</b> .....	69

## Chapitre V: Synthèse des commandes

<b>V.1 Introduction</b> .....	70
<b>V.2 linéarisation tangente</b> .....	70
V.2.1 principe.....	70
V.2.2 points d'équilibre.....	71

V.2.3 Linéarisation locale .....	71
V.2.4 Le modèle linéaire tangent .....	72
<b>V.3 Commandabilité du simulateur .....</b>	<b>73</b>
<b>V.4 Commande multi contrôleurs avec commutateur à événements discrets .....</b>	<b>74</b>
V.4.1 Synthèse du commutateur à événements discret : .....	74
V.4.2 Commande multi contrôleurs par retour d'état .....	78
V.4.3 Commande multi contrôleurs $H_\infty$ .....	80
V.4.4 Commande multi contrôleurs RST .....	87
<b>V.5 Commande multi contrôleurs avec commutateur flou .....</b>	<b>93</b>
V.5.1 Synthèse du commutateur flou .....	93
V.5.2 Commande multi contrôleurs par retour d'état avec commutateur flou .....	95
V.5.3 Commande multi contrôleurs $H_\infty$ avec commutateur flou .....	100
V.5.4 Commande multi contrôleurs RST avec commutateur flou .....	102
<b>V.6 Conclusion .....</b>	<b>105</b>

## Chapitre VI: Implémentation et résultats expérimentaux

<b>VI.1 Introduction .....</b>	<b>107</b>
<b>VI.2 Considérations pratiques .....</b>	<b>107</b>
<b>VI.3 Implémentation des commandes avec commutateur à événements discrets .....</b>	<b>109</b>
VI.3.1 Commande multi contrôleurs $H_\infty$ avec commutation franche .....	109
<i>VI.3.1.1 Stabilisation et poursuite .....</i>	<i>109</i>
<i>VI.3.1.2 Variations paramétriques .....</i>	<i>111</i>
<i>VI.3.1.3 Rejet de perturbations .....</i>	<i>113</i>
VI.3.2 Commande multi contrôleurs RST avec commutation franche .....	114
<i>VI.3.2.1 Stabilisation et poursuite .....</i>	<i>114</i>
<i>VI.3.2.2 Variation paramétriques .....</i>	<i>116</i>
<i>VI.3.2.3 Rejet de perturbations .....</i>	<i>117</i>
<b>VI.4 Implémentation des commandes avec commutateur flou .....</b>	<b>118</b>
VI.4.1 Commande multi contrôleurs $H_\infty$ avec commutation floue .....	118
<i>VI.4.1.1 Stabilisation et poursuite .....</i>	<i>118</i>
<i>VI.4.1.2 Variations paramétriques .....</i>	<i>120</i>
<i>VI.4.1.3 Rejet de perturbations .....</i>	<i>121</i>
VI.4.2 Commande multi contrôleurs RST avec commutation floue .....	122

VI.4.2.1 Stabilisation et poursuite .....	122
VI.4.2.2 Variations paramétriques .....	124
VI.4.2.3 Rejet de perturbations .....	125
<b>VI.5. Superposition des réponses réelle et des réponses théoriques .....</b>	<b>125</b>
<b>VI.6 Commande multi contrôleurs <math>H_\infty</math>-RST .....</b>	<b>128</b>
VI.6.1 Commande multi contrôleurs $H_\infty$ -RST avec commutation franche.....	128
VI.6.1.1 Stabilisation et poursuite .....	128
VI.6.1.2 Variations paramétriques .....	130
VI.6.1.3 Rejet de perturbations .....	131
VI.6.2 Commande multi contrôleurs $H_\infty$ -RST avec commutation floue .....	132
VI.6.2.1 Variations paramétriques .....	134
<b>VI.7 Etude comparative .....</b>	<b>134</b>
<b>VI.8 Conclusion .....</b>	<b>137</b>

## **Conclusion générale**

## **Bibliographie**

## **Annexe A**

## **Annexe B**

## **Annexe C**

## **Annexe D**

## Liste des Figures

<b>Figure I.1:</b> forces agissant sur l'hélicoptère en vol .....	4
<b>Figure I.2 :</b> Effet de couple et rôle du rotor de queue .....	5
<b>Figure I.3 :</b> le simulateur d'hélicoptère TRMS .....	6
(Twin Rotor MIMO System) .....	6
<b>Figure I.4 :</b> Description du TRMS .....	6
<b>Figure I.5 :</b> Vis de fixation (à gauche axe vertical, à droite axe horizontal) .....	7
<b>Figure I.6 :</b> schéma synoptique du fonctionnement du propulseur .....	8
<b>Figure I.7 :</b> encodeurs optiques .....	8
<b>Figure I.8 :</b> La base du TRMS .....	9
<b>Figure I.9 :</b> Branchement des différents connecteurs .....	10
<b>Figure I.10 :</b> Rapid prototyping VS processus de développement traditionnel .....	12
<b>Figure I.11 :</b> processus de génération du code exécutable .....	13
<b>Figure I.12 :</b> différentes cibles d'implémentation .....	14
<b>Figure I.13 :</b> Librairie de la carte d'acquisition .....	14
<b>Figure I.14 :</b> exemple d'implémentation d'un contrôleur .....	15
<b>Figure II.1 :</b> Forces de gravité agissantes sur le TRMS .....	18
<b>Figure II.2 :</b> Moments de la force aérodynamique et de friction dans le TRMS .....	20
<b>Figure II.3 :</b> influence de l'angle d'élévation sur le moment d'inertie .....	23
<b>Figure II.4 :</b> Moment de la force aérodynamique dans le plan horizontal .....	24
<b>Figure II.5 :</b> Circuit de l'un des propulseurs du TRMS .....	26
<b>Figure II.6 :</b> schéma bloc des propulseurs .....	27
<b>Figure II.7:</b> Schéma bloc du TRMS .....	30
<b>Figure II.8 :</b> Schéma bloc du modèle vertical .....	31
<b>Figure II.9 :</b> Schéma bloc du modèle horizontal .....	32
<b>Figure II.10 :</b> approximation par des atan des caractéristiques non linéaires des moteurs (moteur principal (gauche), moteur de queue (droite)) .....	33
<b>Figure II.11 :</b> réponses libres du TRMS avec $\alpha_{v0} = \alpha_{h0} = 0^\circ$ .....	34
<b>Figure II.12 :</b> réponses du TRMS pour $U_v=U_h=0.5$ et $\alpha_{v0} = \alpha_{h0} = 0^\circ$ .....	35
<b>Figure III.1 :</b> schéma de principe de la commande multi contrôleurs (classe indirecte) .....	37
<b>Figure III.2 :</b> schéma de principe de la commande multi contrôleurs (classe directe) .....	38
<b>Figure III.3 :</b> commande multi contrôleurs approche directe avec décision suivant la sortie du système .....	38
<b>Figure III.4 :</b> schéma de principe d'une commutation régie par un système à événement discret .....	39
<b>Figure III.5 :</b> schéma de commande avec entrée de sélection discrète exogène .....	41
<b>Figure III.6 :</b> Evolutions de la consigne (--), de la sortie (--) sans réinitialisation, de la sortie (-*) avec réinitialisation, du mode fonctionnement et de la constante de réinitialisation de la commande multi-contrôleurs réinitialisée .....	43
<b>Figure III.7 :</b> Architecture du bouclage contenant la poursuite de commande .....	44
<b>Figure III.8 :</b> Architecture de commande pour le suivi de la commande active par la commande latente .....	44
<b>Figure III.9 :</b> Principe de la commutation floue .....	46
<b>Figure III.10 :</b> Système flou TS (approche directe) .....	48
<b>Figure IV.1 :</b> structure d'un asservissement classique .....	50
<b>Figure IV.2 :</b> comportement asymptotique des transferts en boucle fermée .....	51
<b>Figure IV.3 :</b> problème $H^\infty$ standard .....	52
<b>Figure IV.4 :</b> exemple de mise en place d'une synthèse $H^\infty$ .....	53
<b>Figure IV.5 :</b> schéma bloc de la commande RST .....	55
<b>Figure IV.6 :</b> schéma équivalent de la commande RST .....	56
<b>Figure IV.7 :</b> Commande RST centralisée .....	59
<b>Figure IV.8 :</b> schéma bloc de la commande par retour d'état avec action intégrale .....	62
<b>Figure IV.9 :</b> exemple d'ensembles considérés en logique booléenne .....	63
<b>Figure IV.10 :</b> exemple d'ensembles considérés en logique floue .....	63

<b>Figure IV.11</b> : structure de base d'un contrôleur flou .....	68
<b>Figure V.1</b> : Grafcet du commutateur .....	76
<b>Figure V.2</b> : schéma d'implémentation sous stateflow .....	77
<b>Figure V.3</b> : réponses du système et signaux de commande pour une stabilisation en (0,0) avec une commande par retour d'état .....	78
<b>Figure V.4</b> : réponses du système et signaux de commande pour une référence sinusoïdale avec une commande multi contrôleurs par retour d'état à commutation franche .....	79
<b>Figure V.5</b> : commutation en tout ou rien des régulateurs par retour d'état pour un signal de référence sinusoïdal .....	80
<b>Figure V.6</b> : tracé des valeurs singulières des matrices fonctions de sensibilité des différents points de fonctionnement .....	81
<b>Figure V.7</b> : tracé des valeurs singulières des matrices fonctions de sensibilité complémentaires des différents points de fonctionnement .....	82
<b>Figure V.8</b> : réponses du système et signaux de commande pour une stabilisation en (0,0) avec une commande $H_\infty$ .....	83
<b>Figure V.9</b> : réponses du système et signaux de commande pour un signal de référence carré avec une commande multi contrôleurs $H_\infty$ à commutation franche .....	84
<b>Figure V.10</b> : commutation en tout ou rien des régulateurs $H_\infty$ pour un signal de référence carré .....	84
<b>Figure V.11</b> : réponses du système et signaux de commande pour un signal de référence en sinusoïde en référence avec une commande multi contrôleurs $H_\infty$ à commutation franche .....	85
<b>Figure V.12</b> : commutation en tout ou rien des régulateurs $H_\infty$ pour un signal de référence en sinusoïde .....	85
<b>Figure V.13</b> : trajectoire dans l'espace d'une poursuite de sinusoïdes avec une commande multi contrôleurs $H_\infty$ à commutation franche .....	86
<b>Figure V.14</b> : Grafcet du commutateur du sous système d'élévation .....	88
<b>Figure V.15</b> : schéma d'implémentation sous stateflow du commutateur du sous système d'élévation .....	88
<b>Figure V.16</b> : réponses du système et signaux de commande pour une stabilisation en (0,0) avec un contrôleur RST .....	89
<b>Figure V.17</b> : réponses du système et signaux de commande pour un signal de référence carré avec une commande multi contrôleur RST à commutation franche .....	89
<b>Figure V.18</b> : commutation en tout ou rien des régulateurs RST pour un signal de référence carré .....	90
<b>Figure V.19</b> : réponses du système et signaux de commande pour un signal de référence sinusoïdal avec une commande multi contrôleurs RST à commutation franche .....	91
<b>Figure V.20</b> : commutation en tout ou rien des régulateurs RST pour un signal de référence sinusoïdal .....	92
<b>Figure V.21</b> : trajectoire dans l'espace d'une poursuite de sinusoïdes avec une commande multi contrôleurs RST à commutation franche .....	93
<b>Figure V.22</b> : Partition floue des variables d'entrée .....	94
<b>Figure V.23</b> : réponses du système et signaux de commande pour un signal de référence sinusoïdal avec une commande multi contrôleurs par retour d'état à commutation floue .....	96
<b>Figure V.24</b> : Fonctions d'activations floues des régulateurs par retour d'état pour une sinusoïde en référence .....	97
<b>Figure V.25</b> : trajectoire du simulateur dans l'espace avec une commande par retour d'état et commutateur flou .....	98
<b>Figure V.26</b> : réponses du système à une référence carrée avec une commande multi contrôleurs par retour d'état à commutation floue .....	99
<b>Figure V.27</b> : Fonctions d'activations floues des régulateurs par retour d'état pour un signal carré en référence .....	99
<b>Figure V.28</b> : réponses du système à une sinusoïde en référence avec une commande $H_\infty$ et commutateur flou .....	100
<b>Figure V.29</b> : Fonctions d'activations floues des régulateurs $H_\infty$ .....	101
<b>Figure V.30</b> : trajectoire du simulateur dans l'espace avec une commande $H_\infty$ et commutateur flou .....	101
<b>Figure V.31</b> : Partition floue relative à l'angle d'élévation .....	102
<b>Figure V.32</b> : réponses du système et signaux de commande pour un signal de référence carré avec une commande multi contrôleur RST à commutation floue .....	103
<b>Figure V.33</b> : Fonctions d'activations floues des régulateurs RST pour un signal de référence carré .....	103
<b>Figure V.34</b> : réponses du système et signaux de commande pour un signal de référence sinusoïdal avec une commande multi contrôleur RST à commutation floue .....	104
<b>Figure V.35</b> : Fonctions d'activations floues des régulateurs RST pour un signal de référence sinusoïdal .....	104
<b>Figure V.36</b> : trajectoire dans l'espace d'une poursuite de sinusoïdes avec une commande multi contrôleurs RST à commutation floue .....	105
<b>Figure VI.1</b> : temps de calcul et période d'échantillonnage .....	107

<b>Figure VI.2</b> : boucle de régulation échantillonnée .....	108
<b>Figure VI.3</b> : initialisation dans la position (0,0) .....	108
<b>Figure VI.4</b> : stabilisation sur une référence carrée avec commande $H^\infty$ CED .....	109
<b>Figure VI.5</b> : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande $H^\infty$ CED .....	110
<b>Figure VI.6</b> : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande $H^\infty$ CED .....	111
<b>Figure VI.7</b> : stabilisation en présence de variations paramétriques avec commande $H^\infty$ CED .....	112
<b>Figure VI.8</b> : poursuite d'une sinusoïde de 0.02 Hz en présence de variations paramétriques avec commande $H^\infty$ CED .....	113
<b>Figure VI.9</b> : rejet de perturbations avec la commande multi contrôleurs $H^\infty$ CED .....	113
<b>Figure VI.10</b> : stabilisation sur une référence carrée par une commande RST CED .....	114
<b>Figure VI.11</b> : poursuite d'une sinusoïde de 0.02 Hz par une commande RST CED .....	115
<b>Figure VI.12</b> : poursuite d'une sinusoïde de 0.04 Hz par une commande RST CED .....	116
<b>Figure VI.13</b> : stabilisation par une commande RST CED en présence de variations paramétriques .....	116
<b>Figure VI.14</b> : poursuite d'une sinusoïde 0.02 Hz par une commande RST CED en présence de variations paramétriques .....	117
<b>Figure VI.15</b> : rejet de perturbations par une commande RST CED .....	117
<b>Figure VI.16</b> : stabilisation sur une référence carrée avec commande $H^\infty$ floue .....	118
<b>Figure VI.17</b> : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande $H^\infty$ floue .....	119
<b>Figure VI.18</b> : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande $H^\infty$ floue .....	120
<b>Figure VI.19</b> : stabilisation en présence de variations paramétriques avec commande $H^\infty$ floue .....	120
<b>Figure VI.20</b> : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande $H^\infty$ floue en présence de variations paramétriques .....	121
<b>Figure VI.21</b> : rejet de perturbations par une commande $H^\infty$ floue .....	121
<b>Figure VI.22</b> : stabilisation sur une référence carrée avec commande RST floue .....	122
<b>Figure VI.23</b> : poursuite d'une sinusoïde de 0.02 Hz par une commande RST floue .....	123
<b>Figure VI.24</b> : poursuite d'une sinusoïde de 0.02 Hz par une commande RST floue en présence de variations paramétriques .....	124
<b>Figure VI.25</b> : rejet de perturbations par une commande RST floue .....	125
<b>Figure VI.26</b> : réponse réelle avec réponse du modèle (commande $H^\infty$ CED) .....	126
<b>Figure VI.27</b> : réponse réelle avec réponse du modèle (commande $H^\infty$ floue) .....	126
<b>Figure VI.28</b> : réponse réelle avec réponse du modèle (commande RST CED) .....	127
<b>Figure VI.29</b> : réponse réelle avec réponse du modèle (commande RST floue) .....	127
<b>Figure VI.30</b> : stabilisation sur une référence carrée avec commande $H^\infty$ -RST SED .....	128
<b>Figure VI.31</b> : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande $H^\infty$ -RST CED .....	129
<b>Figure VI.32</b> : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande $H^\infty$ -RST SED .....	130
<b>Figure VI.33</b> : stabilisation en présence de variations paramétriques avec commande $H^\infty$ -RST SED .....	130
<b>Figure VI.34</b> : rejet de perturbations par une commande $H^\infty$ -RST CED .....	131
<b>Figure VI.35</b> : rejet de perturbations par une commande $H^\infty$ -RST CED lors d'une poursuite de sinusoïde ..	132
<b>Figure VI.36</b> : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande $H^\infty$ -RST floue .....	132
<b>Figure VI.37</b> : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande $H^\infty$ -RST floue .....	133
<b>Figure VI.38</b> : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande $H^\infty$ -RST floue en présence de variations paramétriques .....	134
<b>Figure A.1</b> : diagramme en blocks .....	I
<b>Figure A.2</b> : attribution des pins du connecteur E/S du PCI-1711 .....	II
<b>Figure B.1</b> : graphes expérimentaux représentant la vitesse de rotation du rotor principal en fonction de la tension à l'entrée du rotor (à gauche) et la force aérodynamique engendrée par la rotation du rotor principal (à droite) .....	IV
<b>Figure B.2</b> : approximation polynomiale des caractéristiques du rotor principal .....	IV
<b>Figure B.3</b> : graphes expérimentaux représentant la vitesse de rotation du rotor secondaire en fonction de la tension à l'entrée du rotor (à gauche) et la force aérodynamique engendrée par la rotation du rotor secondaire (à droite) .....	V
<b>Figure B.4</b> : approximation polynomiale des caractéristiques du rotor de queue .....	V
<b>Figure B.5</b> : Structure du superviseur basée sur la prédiction et l'optimisation d'un critère .....	VI
<b>Figure C .1</b> : Réponse du régulateur PID pour une poursuite de sinusoïde de fréquence 0.02 Hz et d'amplitude 30° .....	IX
<b>Figure C .2</b> : Réponse du régulateur PID pour une poursuite d'un signal en créneaux de fréquence 0.02 Hz et d'amplitude 30° .....	X

## Liste des tableaux

<b>Tableau II.1</b> : <i>Les paramètres du modèle</i> .....	33
<b>Tableau IV.1</b> : <i>Représentation d'état des différentes matrices servant au calcul du RST centralisé</i> .....	59
<b>Tableau V.1</b> : <i>points de fonctionnement choisis pour la linéarisation</i> .....	72
<b>Tableau V.2</b> : <i>Base de règles, (3×3)</i> .....	95
<b>Tableau VI.1</b> : <i>Comparaison des performances</i> .....	135
<b>Tableau VI.2</b> : <i>Comparaison de la robustesse en présence de variations paramétriques</i> .....	135
<b>Tableau VI.3</b> : <i>Comparaison des performances vis-à-vis du rejet de perturbation</i> .....	136
<b>Tableau A.1</b> : <i>description des différents signaux de connexion entrée sortie</i> .....	III

## Introduction générale

Suites aux différents progrès technologiques que connaît le monde dans tous les domaines, et particulièrement dans le domaine de l'aéronautique, l'automatique a connu un essor considérable. L'élaboration de lois de commande de systèmes a fait l'objet de nombreux travaux de recherche. L'application de certaines de ces méthodes à des avions et aux hélicoptères a montré leur efficacité. Cependant, les systèmes à commander devenant de plus en plus complexes, les lois de commande deviennent elles-mêmes complexes et parfois difficiles à mettre en œuvre. Pour cela, des prototypes (ou simulateurs) ont été conçus dans le but de tester de nouvelles lois de commandes, qui ne peuvent être implémentées directement sur les systèmes réels. Le simulateur de vol d'hélicoptère *TRMS* (Twin Rotor Mimo System) est l'un de ces prototypes. Ce simulateur est un système multi variables à deux entrées et deux sorties, caractérisé par une forte non linéarité, un couplage important et une dynamique instable en boucle ouverte.

D'un autre côté, l'obtention d'un modèle précis du procédé devient nécessaire. Les méthodes de modélisation traditionnelles s'avèrent souvent incapables de représenter le comportement global d'un système. L'automaticien est alors confronté à un problème de choix pour la construction d'une loi de commande.

Dans le cadre du présent travail, il s'agit de s'intéresser à la commande du simulateur *TRMS*. Deux hypothèses de base concernant le système sont associées à ce travail. La première repose sur le fait que les non-linéarités sont suffisamment importantes pour qu'un contrôleur linéaire ne puisse pas donner d'assez bonnes performances. En effet, si cette hypothèse est effectivement respectée, quel que soit le degré de sophistication des régulateurs conventionnels linéaires, leurs performances sont largement détériorées en présence de variations paramétriques ou de perturbations caractérisant de tels systèmes. L'utilisation d'un unique régulateur linéaire est écartée. La deuxième hypothèse est basée sur le fait que nous disposons d'un minimum d'informations sur le comportement du système. En fait, nous sommes en mesure de connaître uniquement son comportement local.

Les commandes étudiées ici sont de type **multi-contrôleurs**. Elles permettent la commutation de lois de commande selon l'environnement du procédé. Elles autorisent l'utilisation de contrôleurs obtenus par des méthodes diverses. Elles sont conçues en deux étapes :

- Synthèse de contrôleurs locaux : cette première étape se sert de la deuxième hypothèse citée précédemment. En effet, connaissant a priori le comportement du système dans différentes zones de fonctionnement, la synthèse de contrôleurs locaux devient alors aisée.
- L'étape suivante est de prendre en compte ces lois de commande locales afin d'élaborer la loi de commande effectivement appliquée au procédé. Ce rôle est rempli par le commutateur éventuellement contrôlé par un superviseur. La synthèse d'un commutateur et d'un superviseur est une étape importante.

Ce travail comporte six chapitres :

Dans le **premier chapitre**, nous présentons le système en question, ses constituants, son principe de fonctionnement, et l'environnement de développement.

Le **second chapitre** sera consacré à une modélisation analytique du TRMS, suivie de quelques simulations en boucle ouverte.

Dans le **troisième chapitre**, nous présentons quelques approches de la commande multi contrôleurs. Deux types de commutateurs seront étudiés : le premier consiste à sélectionner un seul contrôleur parmi un ensemble de contrôleurs locaux (commutation franche), le deuxième est nommé "commutateur flou". Ce dernier permet de pondérer les différents signaux de commande pour élaborer le signal de commande final.

Dans le **quatrième chapitre**, nous présentons un bref rappel théorique sur la commande par retour d'état, la commande  $H_\infty$  et la commande RST.

Le **cinquième chapitre**, concerne la synthèse des lois de commande et les simulations correspondantes.

Le **sixième chapitre** évoquera les considérations pratiques. Les résultats expérimentaux de l'implémentation des lois de commande synthétisées sur le simulateur seront présentés puis discutés selon leurs performances et robustesses.

Ce travail sera clôturé par une conclusion générale.

# Chapitre I

## *Présentation du simulateur d'hélicoptère*

### **I.1 Introduction**

Les systèmes aéronautiques relèvent de processus suffisamment complexes et onéreux au point où il n'est plus possible d'implémenter des commandes directement sur ceux-ci. D'un autre côté, le développement de l'électronique de puissance permet l'élaboration de systèmes miniatures. En effet, plusieurs prototypes sont mis en œuvre pour tester les commandes avant de les transférer sur le système réel. Le simulateur d'hélicoptère (TRMS) est l'un de ces prototypes qui supporte l'implémentation des diverses commandes. Il sera donc l'objet de notre travail.

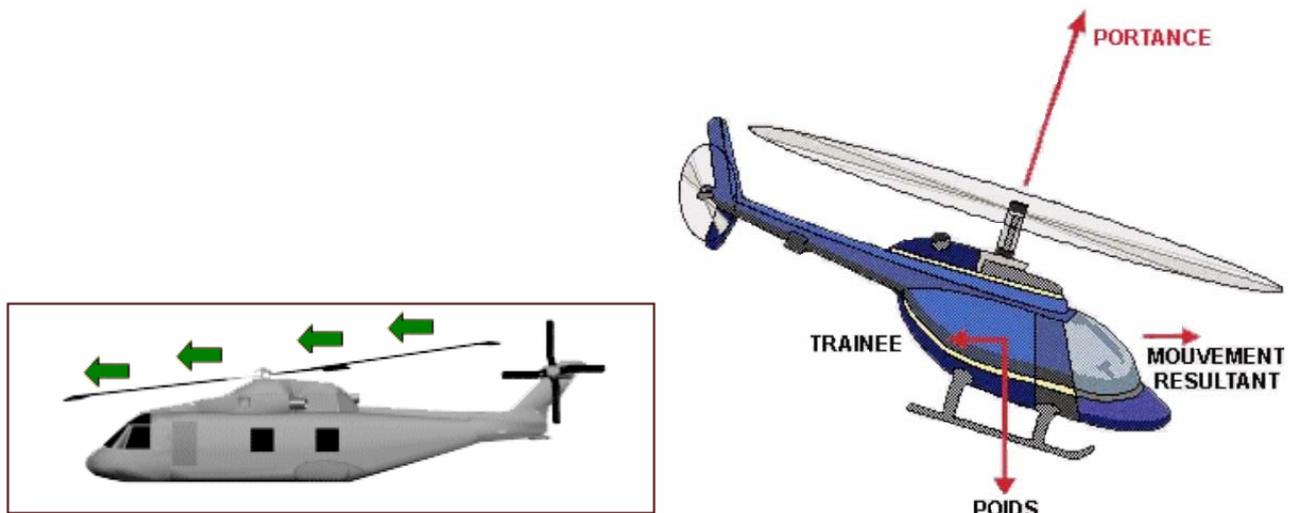
Ce chapitre abordera un petit aperçu sur le principe du vol des hélicoptères, ensuite seront présentés les détails du simulateur.

### **I.2 Principe de vol d'un hélicoptère**

Essayons de simplifier le principe de vol d'un hélicoptère en nous référant à l'avion. Dans un avion, la traction est assurée par les hélices qui font prendre de la vitesse à l'avion. Quant à la portance, ce sont les ailes qui, par leurs formes aérodynamiques, créent une force proportionnelle (au carré généralement) de la vitesse de traction provoquant, ainsi, l'envol de l'avion, dès que sa vitesse est suffisamment importante.

Pour un hélicoptère, c'est le rotor qui joue à la fois le rôle de l'aile et de l'hélice. Il assure, ainsi, traction et portance.

Ainsi, Pour changer de direction l'hélicoptère doit changer l'angle d'attaque de son rotor pour créer une portance dans la direction voulue.



**Figure I.1:** forces agissant sur l'hélicoptère en vol

Le rotor est entraîné en rotation par un groupe motopropulseur, constitué d'un moteur (à pistons ou turbine à gaz), d'un embrayage et d'un réducteur aussi appelé BTP (boîtier de transmission principal).

Cependant, comme pour l'hélice, le rotor exerce sur l'air un couple dû à son mouvement. L'air en retour (principe d'action-réaction), exerce un autre couple d'intensité égale mais de sens opposé. Imaginez que vous essayez d'introduire une vis encastrée dans un mur extrêmement dur avec une visseuse puissante, vous tourneriez avec la visseuse ! C'est le même phénomène qui est observé avec le rotor principal.

Pour contrer cet effet de couple et permettre un vol stable, on dispose à l'arrière des hélicoptères un "rotor de queue". Son rôle consiste à contrer le couple du rotor principal.

C'est le système le plus couramment utilisé actuellement mais il existe d'autres systèmes qui jouent le même rôle tel que : jet d'air, fenestron, deux rotors principaux (ce qui annule l'anti-couple), mais aussi deux rotors d'anti-couple.

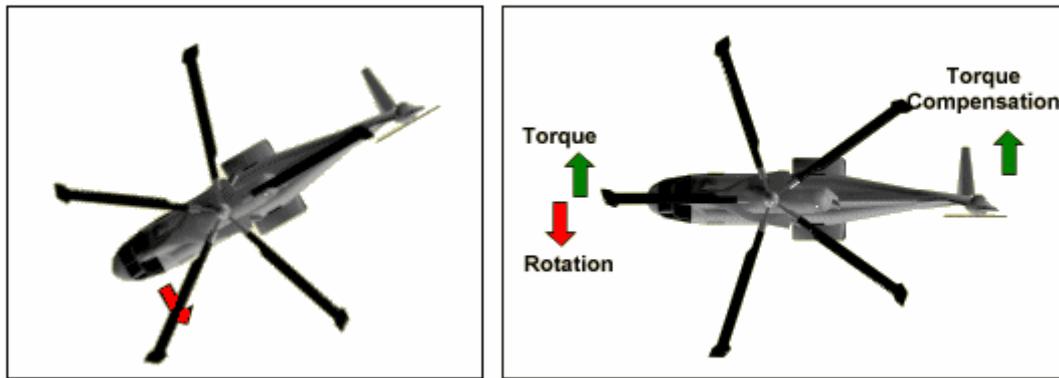


Figure I.2 : Effet de couple et rôle du rotor de queue

Le rotor anti-couple consomme environ 10 % de la puissance, son interaction sonore avec le rotor principal le rend bruyant, dangereux (extrêmement coupant), sensible au vent de travers, il est également coûteux et fragile, toutes ces raisons motivent les constructeurs à l'éviter ou à l'optimiser.

### I.3 Description du simulateur [1], [3]

Le TRMS (Twin Rotor Mimo System) est un dispositif de laboratoire conçu dans le but d'expérimenter de nouvelles lois de commandes. Son comportement est similaire à celui d'un hélicoptère. Du point de vue commande ; c'est un exemple de système présentant une non linéarité d'ordre supérieur fortement couplé.

Le TRMS est formé de (figure I.3):

- une poutre pouvant pivoter sur sa base de manière à ce qu'elle puisse tourner librement dans les plans vertical et horizontal ;
- Deux propulseurs (principal et secondaire) fixés aux deux extrémités de la poutre. Ils sont formés d'une hélice, d'un moteur à courant continu et d'un bouclier de sécurité ;
- Un contrepoids fixé sur une tige collée au niveau du pivot de la poutre, son rôle est de diminuer les vibrations (oscillations) de cette dernière ;
- Une tour pour maintenir la poutre ;
- Une base comprenant des circuits électroniques pour l'adaptation, la synchronisation et le filtrage des signaux entrants et sortants ;
- Un boîtier de marche/arrêt des moteurs.

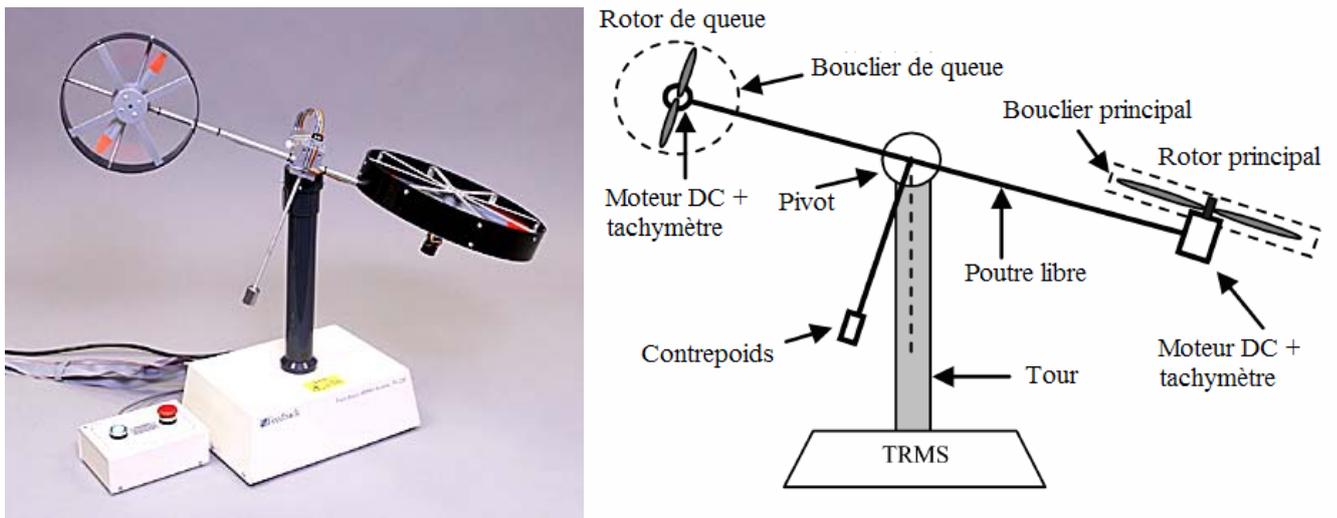


Figure I.3 : le simulateur d'hélicoptère TRMS

(Twin Rotor MIMO System)

L'état de la poutre est décrit par 4 variables : les angles horizontal (azimut) et vertical (élévation) mesurés à l'aide d'un capteur se trouvant au niveau du pivot (et qui représente également les sorties du système).

Les deux autres variables sont les vitesses angulaires des rotors, mesurées par des générateurs tachymétrique couplés aux moteurs DC. Elles ne sont, néanmoins, pas utilisées dans les commandes.

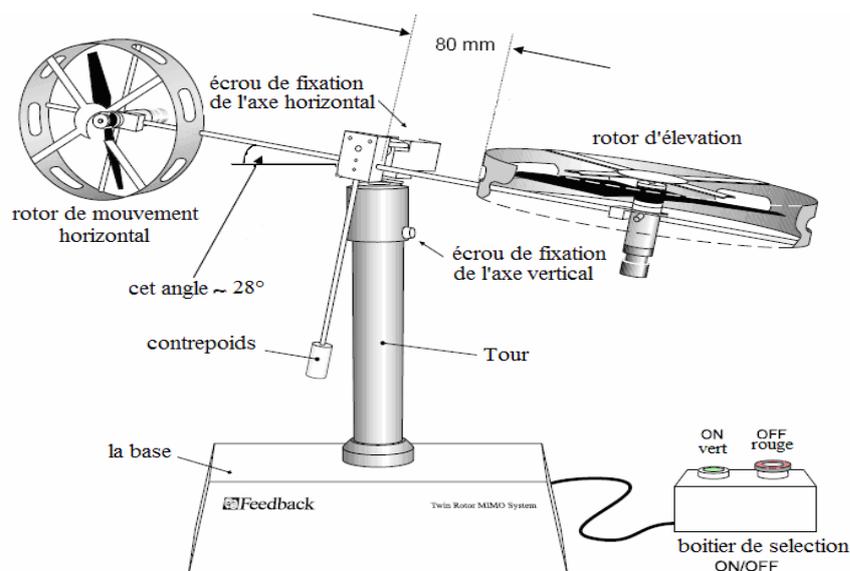
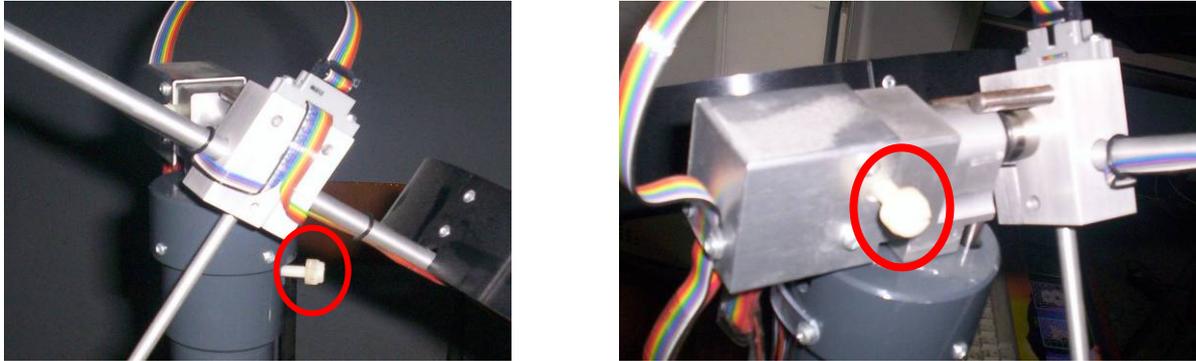


Figure I.4 : Description du TRMS

Il existe un fort couplage entre les actions des rotors, chaque rotor influence les deux angles.

L'articulation sphérique permet à la poutre de pivoter simultanément dans le plan horizontal et vertical. C'est un système à deux degrés de liberté, mais on peut restreindre le mouvement à un degré de liberté à l'aide de 2 vis. (Figure I.5)



**Figure I.5 :** *Vis de fixation (à gauche axe vertical, à droite axe horizontal)*

Dans un hélicoptère réel, la force aérodynamique est contrôlée par changement de l'angle d'attaque. Cependant, pour le TRMS, l'angle d'attaque étant fixe, la force aérodynamique est commandée par changement de la vitesse des rotors. On utilise donc les tensions appliquées aux moteurs comme variable de commande.

### **I.3.1 Propulseurs**

Le TRMS possède deux propulseurs, un principal qui agit dans le plan vertical, et l'autre secondaire qui agit dans le plan horizontal. Chaque propulseur est composé d'une hélice. Ces dernières sont entraînées par des moteurs à courant continu (marque MAXON), couplés avec des tachymètres. La tension de commande varie entre  $\pm 2.5V$ , le signe de la tension indique le sens de rotation.

### **I.3.2 Commande des moteurs**

Les moteurs électriques à courant continu offrent l'avantage de présenter des vitesses de rotation proportionnelles à la tension qui leur est appliquée. Le mécanisme utilisé est un hacheur, qui est un convertisseur statique alimenté par une source de tension continue produisant aux bornes du moteur une tension de valeur moyenne réglable. Ce hacheur est

commandé par des trains d'impulsions (MLI) qui sont générés à partir du signal de commande provenant du PC à travers la carte d'acquisition.

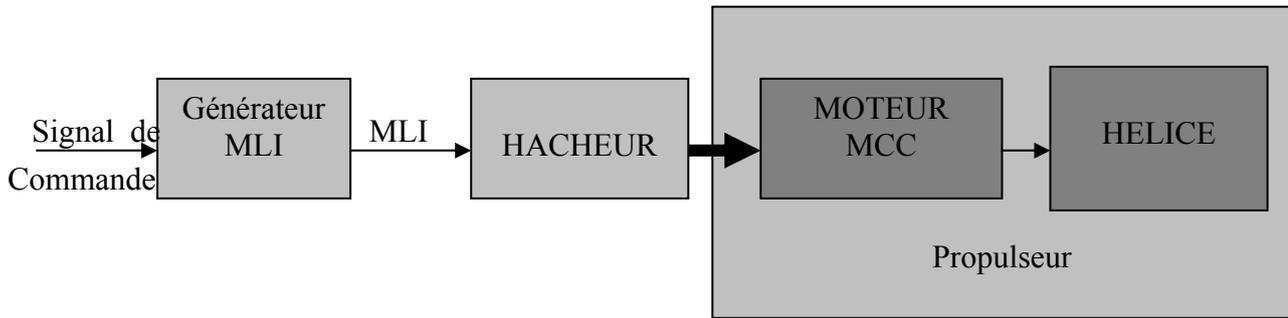


Figure I.6 : schéma synoptique du fonctionnement du propulseur

### I.3.3 Les encodeurs optiques [4]

Ces capteurs servant à la mesure des angles d'azimut et d'élévation se situent à l'intérieur du pivot sphérique de la poutre. Il s'agit de deux encodeurs optiques incrémentaux dont le principe de fonctionnement est le suivant [figure I.6]:

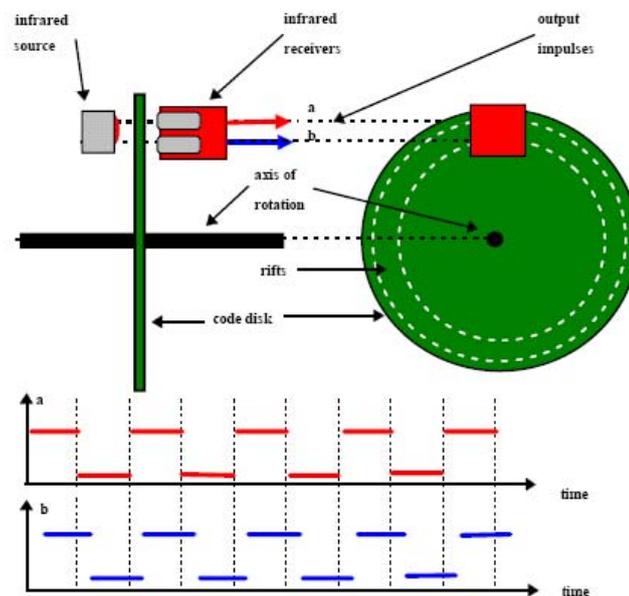


Figure I.7 : encodeurs optiques

Les faisceaux lumineux émis par les deux photodiodes (A et B) passent à travers deux anneaux de fentes sur le disque. Les fentes sont décalées, de sorte que les tensions des

récepteurs (A et B) soient des ondes rectangulaires en quadrature de phase. Le dispositif ainsi réalisé permet de déterminer le sens de rotation.

### I.3.4 La base du TRMS [3]

À l'arrière de la base se trouvent les différentes connectiques (figure I.7):

- Connecteur ON/Off : fiche DIN pour connecter la boîte de l'intercepteur ON/OFF des moteurs ;
- Sélecteur 110/220 V ;
- Fiche d'alimentation 110/220 V ;
- **Connecteur CN1** : il se branche avec le PL1 de la boîte d'adaptation, avec une large nappe de 40 pistes. Les signaux sont ceux des deux capteurs de position, ce sont des signaux digitaux sur 16 bits ;
- **Connecteur CN2** : il se branche avec PL3, avec une nappe de 20 pistes. Les signaux sont les tensions de commandes des deux moteurs ;
- **Connecteur CN3** : il se branche avec PL2, avec une nappe de 20 pistes. Les signaux sont ceux des tachymètres.

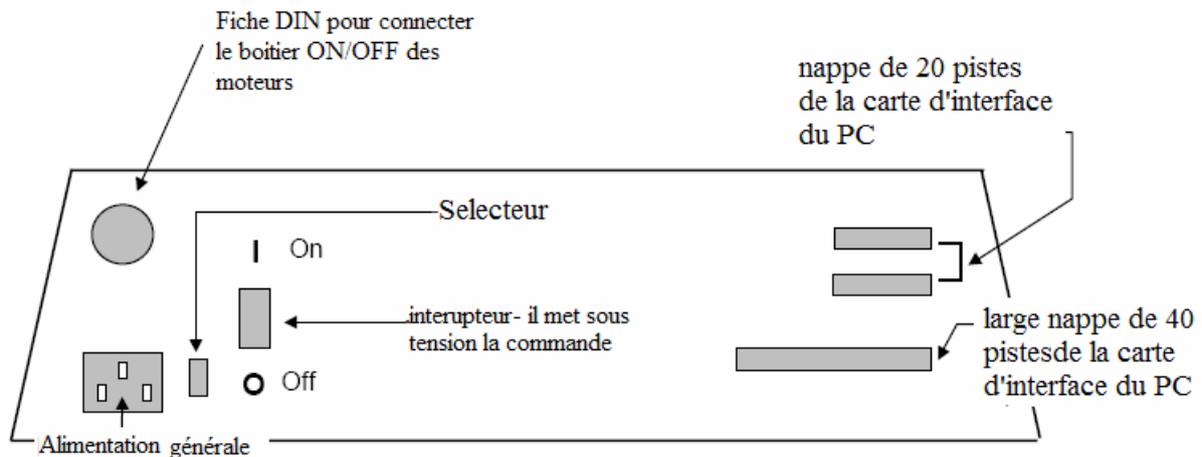
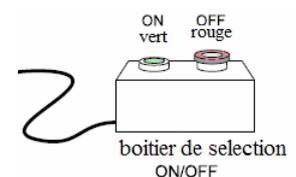


Figure I.8 : La base du TRMS

### I.3.5 Boîtier marche/arrêt

C'est un petit boîtier avec deux boutons poussoirs, l'un vert pour fermer le circuit d'amplification des moteurs, et l'autre rouge pour couper l'alimentation. Son rôle principal consiste à protéger les moteurs en cas de mauvaise manœuvre. Il est connecté à l'arrière de la base du TRMS.



### I.3.6 La SCSI BOX

La SCSI BOX (figure I.8.b) a comme rôle d'adapter les signaux entre la carte d'acquisition et les différents connecteurs de la base du *TRMS* et de séparer ces différents signaux en fonction de leurs types. La boîte dispose de quatre connecteurs :

- un connecteur 68 pin pour connecter le câble global 68 pin SCSI de la carte d'acquisition ;
- un grand connecteur PL1 pour les entrées digitales des capteurs de positions (deux encodeurs optiques) ;
- un petit connecteur PL2 pour les sorties analogiques (deux tensions de commande des moteurs) ;
- un autre petit connecteur PL3 pour les entrées analogiques (retour tachymétrique des deux moteurs).

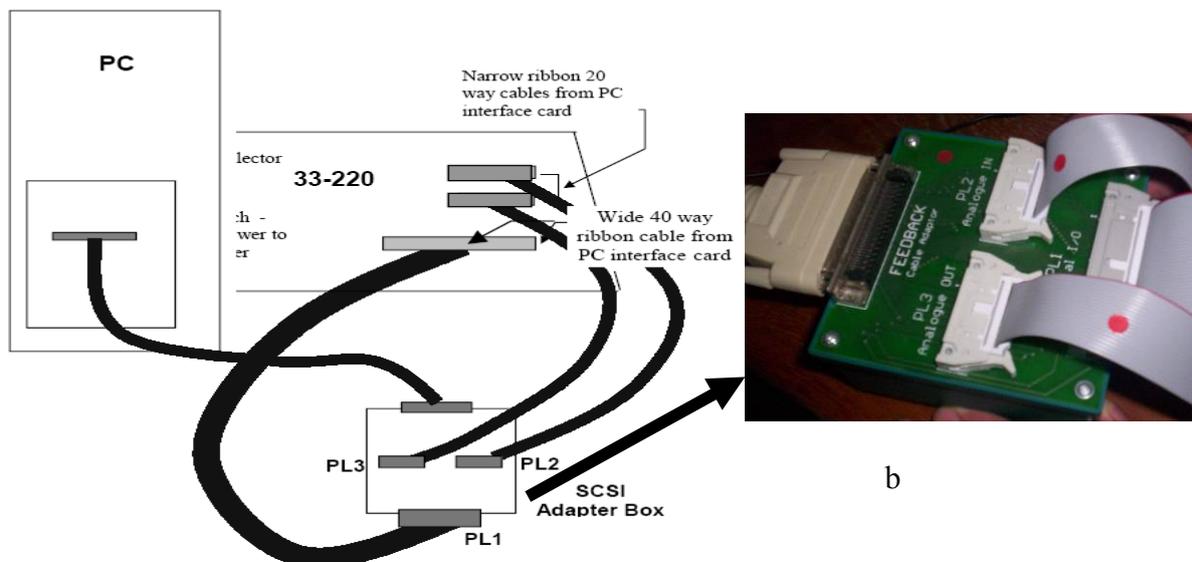


Figure I.9 : Branchement des différents connecteurs

### I.3.7 Carte d'acquisition ADVANTECH PCI 1711 [pci1711]

C'est une carte d'acquisition universelle qui s'installe sur le port PCI du PC de commande et dispose de connecteurs extérieurs pour des entrées/sorties analogiques et digitales. Ces principales caractéristiques sont :

- La fonction Plug & play ;

- 16 entrées analogiques configurables 16 simples (single-ended) ou source flottante ;
- Convertisseurs A/D industriels normalisés à approximations successives ;
- 12 bits utilisés pour la conversion des entrées analogiques ;
- La fréquence maximale d'échantillonnage est de 100KHz ;
- Gamme des entrées analogiques programmables et contrôlable par software ;
- Chaque canal dispose de sa gamme individuelle stockée dans la RAM de la carte ;
- 2 sorties analogiques (convertisseur D/A) ;
- 16 canaux d'entrées digitales ;
- 16 canaux de sorties digitales ;
- Un compteur/timer programmable ;
- Scan automatique des gains/canaux.

Son principal rôle dans notre application est de permettre la commande digitale d'un système continu à travers un PC, en convertissant les signaux analogiques en numériques et vice-versa. Dans notre cas, on utilise :

- Deux sorties analogiques (convertisseur D/A) pour délivrer les tensions de commande de références ;
- Les entrées digitales des deux encodeurs optiques pour mesurer l'angle d'élévation et d'azimut de la poutre ;
- Deux entrées analogiques (convertisseur A/D) pour récupérer le retour tachymétrique des deux moteurs.

#### **I.4 Environnement de développement [5], [8]**

Ce banc d'essai utilise une approche novatrice d'implémentation, qui jouit d'une flexibilité accrue et d'une architecture totalement ouverte. Cette approche est appelée HIL pour « Hardware-In-the-Loop » ou bien « Rapid Prototyping ».

Cette approche élimine l'étape entre la synthèse du régulateur en simulation et son implémentation réelle sur un système embarqué spécialisé. Cette dernière provoque des pertes de temps considérables, en testant et en calibrant à nouveau le régulateur implémenté. Voici, ci-dessous, un comparatif succinct entre cette approche et l'approche classique :

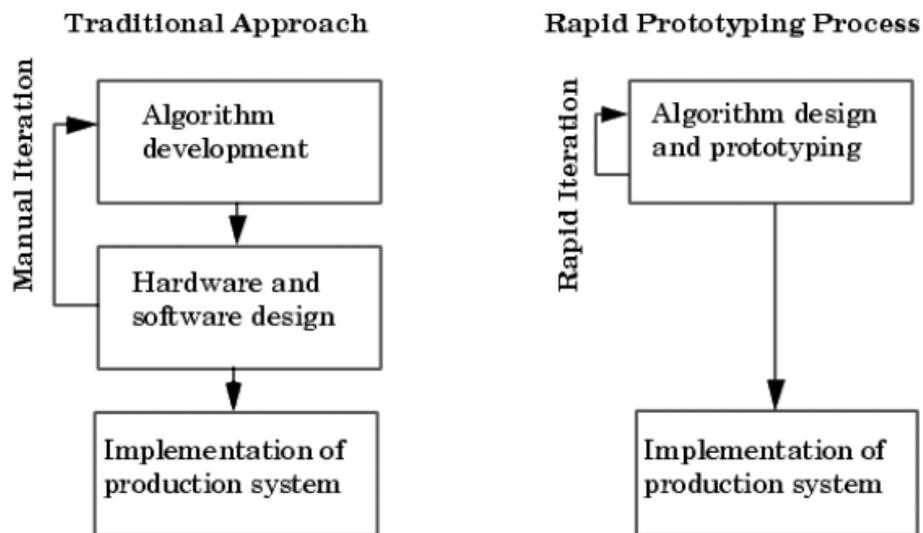


Figure I.10 : *Rapid prototyping VS processus de développement traditionnel*

Avec le « hardware-in-the-loop », on teste directement le contrôleur synthétisé en simulation sur le système physique réel, à l'aide d'un simple PC de développement. En effet, sont nécessaire :

- un PC standard (pour le développement et pour la commande) ;
- une carte d'acquisition pour la conversion A/D et D/A ;
- le système physique à commander ;
- les logiciels appropriés.

Les logiciels utilisés sont :

- MATLAB : il joue le rôle d'une plateforme, où tous les autres composants s'exécutent. C'est l'environnement le plus utilisé dans le monde par les ingénieurs, il fournit plusieurs routines sophistiquées de calculs numériques ;
- SIMULINK : C'est un langage de haut niveau graphique, à l'aide duquel on programme avec des objets ou blocks. Ces objets peuvent être soit des block standards ou bien des blocks spécialisées conçus par l'utilisateur (ex : les S-functions) ;
- REAL TIME WORKSHOP : c'est le programme le plus important, il génère automatiquement du code source C++ optimisé, à partir du modèle simulink. De plus, ce code est paramétrable pour plusieurs targets (cible) d'implémentation. Dans notre

cas le target est un processeur Pentium sous WINDOWS comme système d'exploitation ;

- Compilateur C++ : il compile le code généré par real time workshop et fait l'édition des liens pour générer un exécutable qui communique avec la cible d'implémentation (target), c'est le compilateur VISUAL C++ 6.0 PRO ;
- REAL TIME WINDOWS TARGET : Dans le cas d'un environnement WINDOWS, ce petit kernel (noyau), est essentiel. Son rôle est primordial, il assure le temps réel pour notre régulateur car WINDOWS est un OS événementiel. de ce fait, il ne permet pas l'exécution en temps réel. Il s'exécute comme un service résident dans le niveau zéro (mode noyau) et intercepte les interruptions matérielles avant WINDOWS. Ensuite, il communique avec l'exécutable du régulateur et joue le rôle d'interface avec le système physique à travers la carte d'acquisition. Il contrôle, donc, le flux de données et de signaux du modèle (maintenant exécutable) vers le système physique et vice versa ;
- On a besoin, également, des bibliothèques, ou drivers de la carte d'acquisition, pour les intégrer dans notre programme

Voici, ci-dessous, un schéma (figure I.11) qui illustre la relation qui existe entre les différents modules :

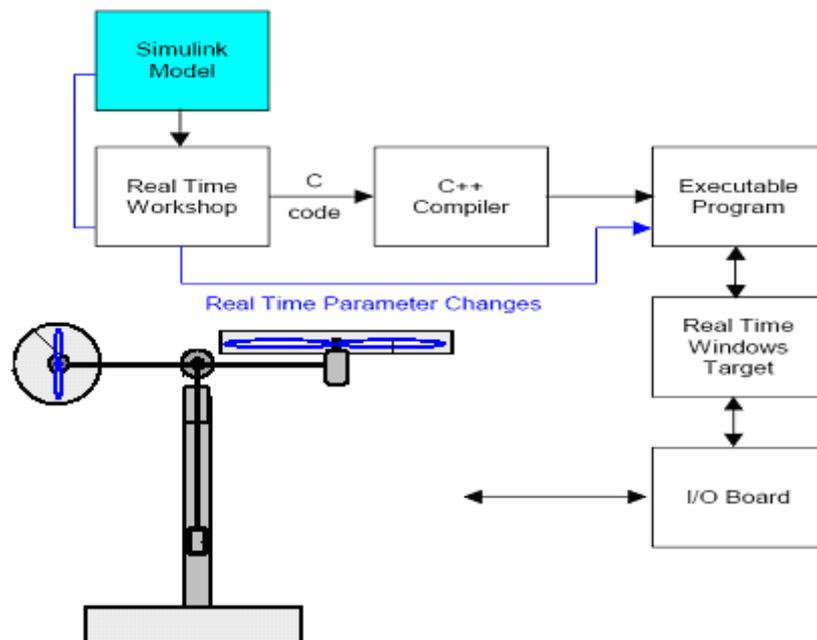


Figure I.11 : processus de génération du code exécutable

Et voici un autre schéma (figure I.12) qui illustre les différentes cibles (targets) et les différentes manières dont on peut implémenter notre régulateur :

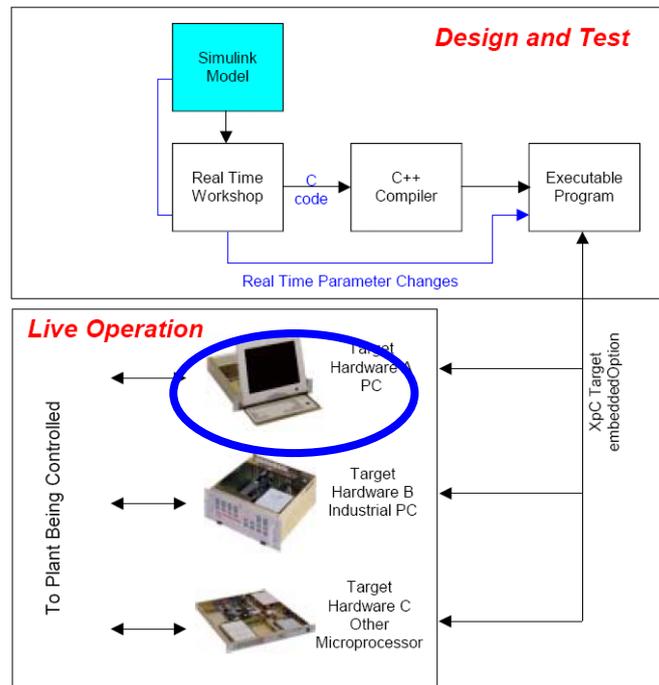


Figure I.12 : différentes cibles d'implémentation

Pour passer d'un régulateur simulé (non temps réel) vers un programme temps réel, il suffit uniquement d'ajouter dans le modèle de simulation sous SIMULINK, les différents blocs des entrées/sorties de la carte d'acquisition (ANALOG INPUT, ANALOG OUTPUT, ENCODER CHANNEL ...etc).

Pour notre carte d'acquisition, la librairie utilisée est la suivante :

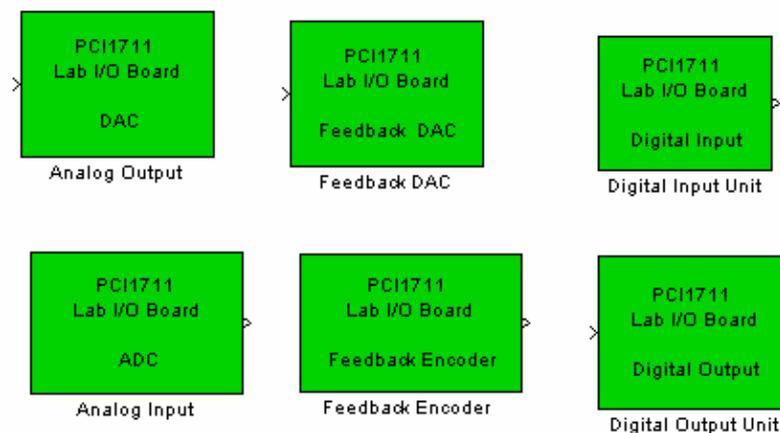


Figure I.13 : Librairie de la carte d'acquisition

Et voici, ci-dessous, un exemple complet de commande en temps réel avec MATLAB :

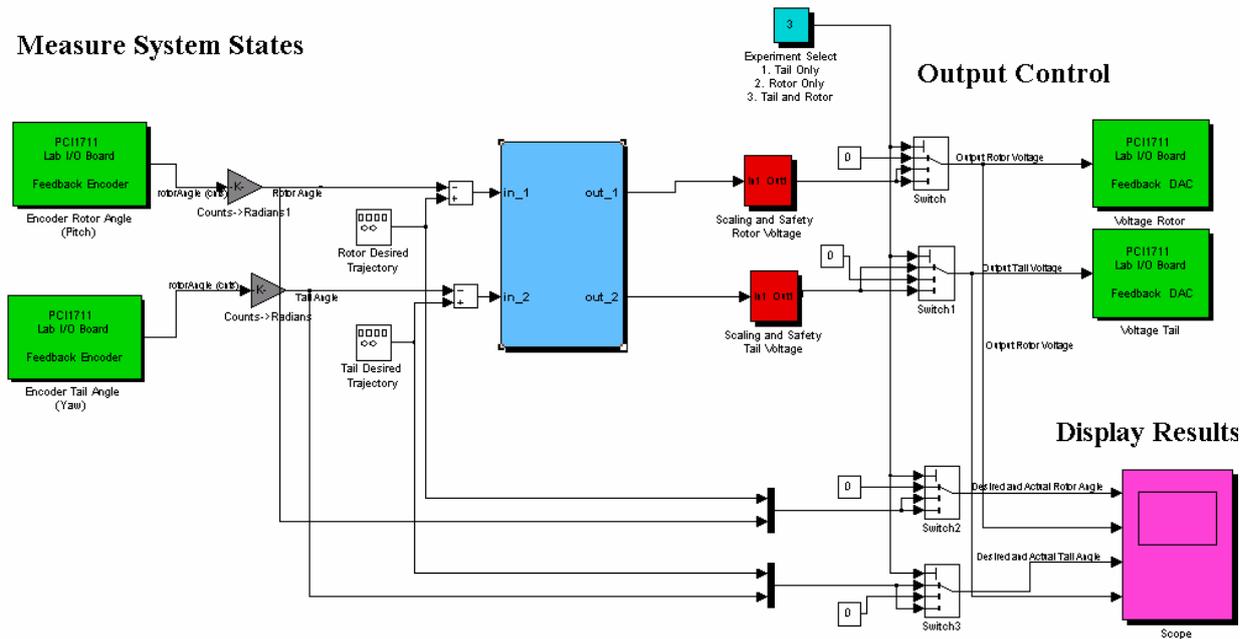


Figure I.14 : exemple d'implémentation d'un contrôleur

**En vert** : blocs de communications avec la carte d'acquisition, à gauche les encodeurs et à droite des sorties analogiques (tension de commande).

**En rouge** : blocs de protection des moteurs, car la tension max est de  $\pm 2.5 V$ .

**En gris** : blocs de conversion, car les encodeurs incrémentaux donnent le nombre d'impulsions qu'il faut convertir en radian.

**En bleu** : est la partie essentielle qui est le bloc de notre régulateur. Ce bloc peut être écrit dans n'importe quel langage, on peut même importer un régulateur déjà écrit et l'implémenté, grâce à l'architecture ouverte de cette approche.

Après compilation du programme, il sera indépendant de MATLAB et SIMULINK, néanmoins, on peut utiliser uniquement SIMULINK comme interface homme/machine, pour visualiser et récupérer les signaux.

## I.5 Conclusion

Dans ce chapitre, nous avons d'abord donné le principe du vol d'un hélicoptère réel. Ensuite nous avons fait une brève description du simulateur (TRMS), ainsi que son principe de fonctionnement. Enfin, on a expliqué l'environnement de développement utilisé.

# Chapitre II

## *Modélisation du TRMS*

### II.1 Introduction

L'analyse du comportement dynamique d'un système passe nécessairement par la recherche de l'ensemble des relations entre les grandeurs d'entrée et de sortie de celui-ci.

Cet ensemble constitue un modèle mathématique exprimant la dynamique du système sous forme d'équations différentielles ordinaires, à dérivées partielles, ou d'équations aux différences.

La construction d'un modèle mathématique peut s'effectuer par application des lois usuelles de la physique, ce modèle est alors dit « modèle de connaissance ». Cependant, en automatique, il arrive que, pour un manque d'informations disponibles sur le système, on se contente de reproduire le comportement entrée/sortie du système à partir de données expérimentales en considérant ce dernier comme une boîte noire. On parle alors d'identification.

On désire, toutefois, que le modèle puisse prévoir au mieux le comportement du système aux diverses excitations (commandes, perturbations ...). Ainsi, plus il est détaillé plus il est fidèle au système. Néanmoins, cela engendre une complication de l'étude et de la synthèse d'éventuelles lois de commande. Un compromis doit alors être fait en passant par des hypothèses simplificatrices afin de pouvoir répondre aux contraintes pratiques.

Dans ce chapitre, nous commencerons par une modélisation détaillée du simulateur présenté dans le chapitre précédent, on présentera ensuite les modèles découplés et on terminera par quelques simulations en boucle ouverte.

## II.2 Modèle non linéaire [4]

Les méthodes de modélisation classiques des systèmes couplés, non linéaires et d'ordre élevé, basées sur les équations de Lagrange, conduisent dans la majorité des cas à des modèles complexes. Pour cela, on utilisera les équations fondamentales de la dynamique (Lois de Newton).

Le modèle physique est développé sous certaines hypothèses simplificatrices :

- la dynamique des sous-systèmes de propulsion peut être décrite par des équations différentielles du premier ordre ;
- les frottements dans le système sont du type visqueux ;
- les sous-systèmes hélices-air peuvent être décrits par les postulats de la théorie de l'écoulement aérodynamique.

### II.2.1 Modélisation du sous système d'élévation

Considérons le mouvement de la poutre uniquement dans le plan vertical (rotation autour de l'axe horizontal), la seconde loi de Newton donne :

$$M_v = J_v \frac{d^2 \alpha_v}{dt^2} \quad (\text{II.1})$$

Avec :

$$M_v = \sum_{i=1}^4 M_{vi} \quad (\text{II.2})$$

$$J_v = \sum_{i=1}^8 J_{vi} \quad (\text{II.3})$$

Ainsi l'équation (II.1) peut être mise sous la forme :

$$J_v \ddot{\alpha}_v = M_{v1} + M_{v2} + M_{v3} + M_{v4} \quad (\text{II.4})$$

Où :

$M_v$  : La somme des moments dans le plan vertical.

$J_v$  : La somme des moments d'inertie par rapport à l'axe horizontal.

$\alpha_v$  : L'angle d'élévation de la poutre.

$M_{v1}$  : Moment de la gravitation.

$M_{v2}$  : Moment de la force aérodynamique.

$M_{v3}$  : Moment de la force centrifuge.

$M_{v4}$  : Moment de friction.

### II.2.1.1 Moment de la gravitation $M_{v1}$

La figure suivante montre l'ensemble des forces de gravité qui agissent sur le sous-système d'élévation. Le moment gravitationnel résultant est la somme des moments de chaque force.

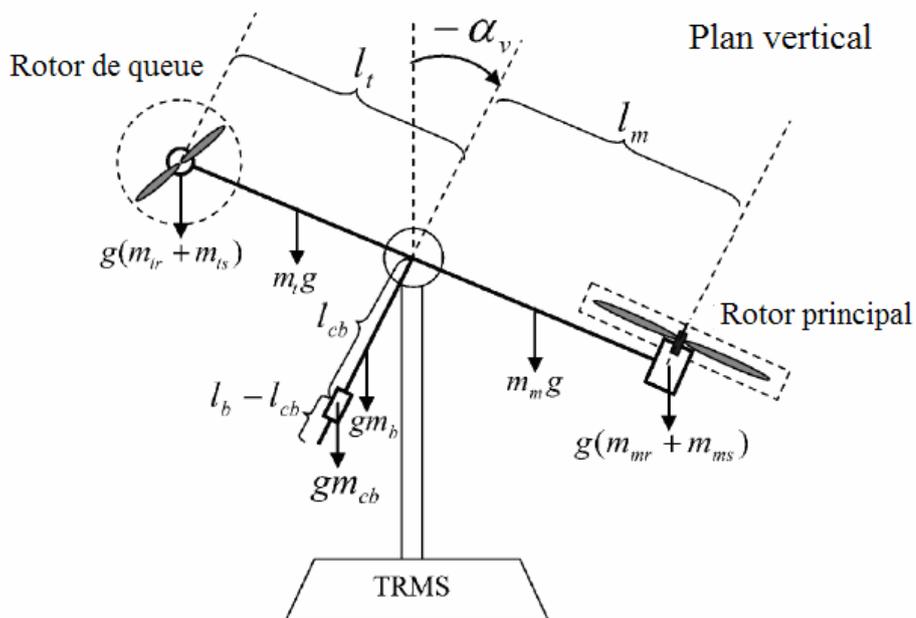


Figure II.1 : Forces de gravité agissantes sur le TRMS

$$M_{v1} = g \left\{ \left[ \left( \frac{m_t}{2} + m_{tr} + m_{ts} \right) l_t - \left( \frac{m_m}{2} + m_{mr} + m_{ms} \right) l_m \right] \cos \alpha_v - \left( \frac{m_b}{2} l_b + m_{cb} l_{cb} \right) \sin \alpha_v \right\} \quad (\text{II.5})$$

Ecriture sous forme condensée :

$$M_{v1} = g \{ [A - B] \cos \alpha_v - C \sin \alpha_v \} \quad (\text{II.6})$$

$$\text{Avec } \begin{cases} A = \left( \frac{m_t}{2} + m_{tr} + m_{ts} \right) l_t \\ B = \left( \frac{m_m}{2} + m_{mr} + m_{ms} \right) l_m \\ C = \left( \frac{m_b}{2} l_b + m_{cb} l_{cb} \right) \end{cases}$$

Où :

$M_{v1}$  : Le moment résultant correspondant aux forces de gravités.

$m_{mv}$  : La masse du rotor principal.

$m_m$  : La masse de la partie principale de la poutre.

$m_{tr}$  : La masse du rotor secondaire.

$m_t$  : La masse de la partie secondaire de la poutre.

$m_{cb}$  : La masse du contrepoids.

$m_b$  : La masse de la poutre du contrepoids.

$m_{ms}$  : La masse de l'hélice principale.

$m_{ts}$  : La masse de l'hélice secondaire.

$l_m$  : La longueur de la partie principale de la poutre.

$l_t$  : La longueur de la partie secondaire de la poutre.

$l_b$  : La longueur de la poutre du contrepoids.

$l_{cb}$  : La distance entre le contrepoids et l'articulation.

$g$  : L'accélération gravitationnelle.

### II.2.1.2 Moment de la force aérodynamique $M_{v2}$

La figure II.2 ci-dessous montre la direction de la force aérodynamique de portance dans le plan vertical. Cette force dépend de la vitesse angulaire du rotor principal dont la relation est déterminée expérimentalement.

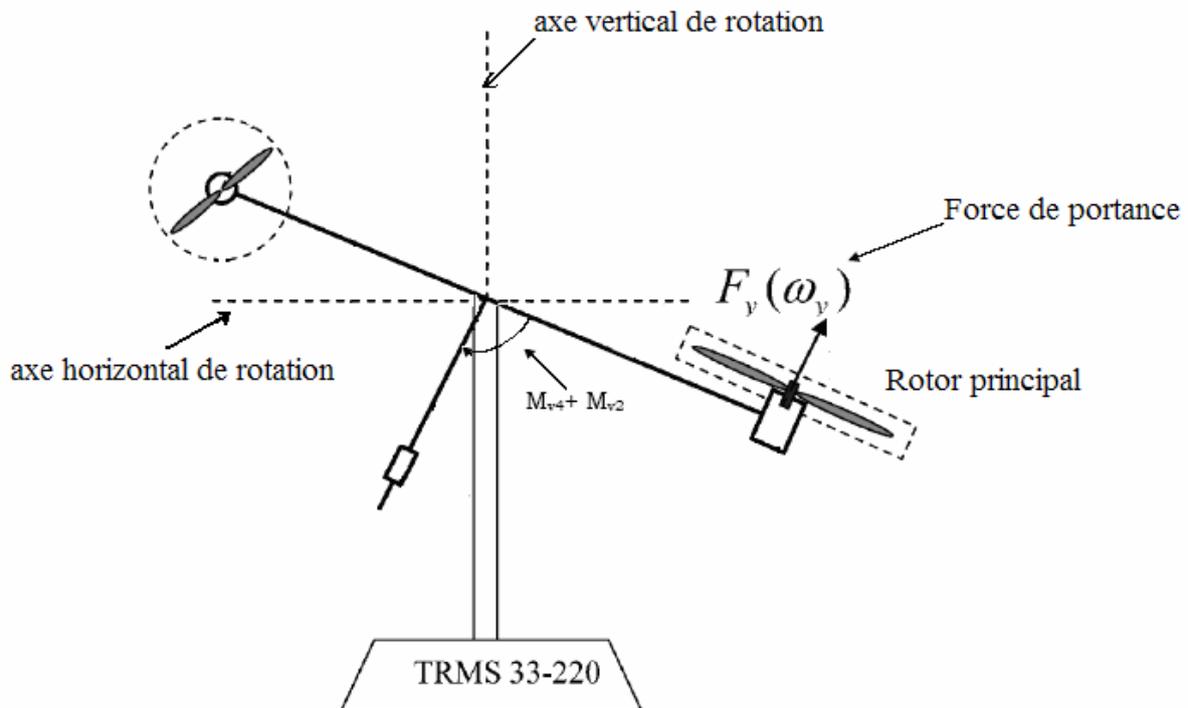


Figure II.2 : Moments de la force aérodynamique et de friction dans le TRMS

$$M_{v2} = l_m F_v(\omega_m) \quad (\text{II.7})$$

Où :

$M_{v2}$  : Le moment de la force aérodynamique développé par le rotor principal.

$\omega_m$  : La vitesse angulaire du rotor principal.

$F_v(\omega_m)$  : Exprime la dépendance de la force aérodynamique de la vitesse angulaire du rotor principal.

### II.2.1.3 Moment de la force centrifuge $M_{v3}$

La masse de la poutre lorsqu'elle effectue un mouvement de rotation dans le plan horizontal génère une force orthogonale à la direction de rotation. Cette force est proportionnelle au carré de la vitesse angulaire de la poutre dans le plan horizontal, elle est

nulle lorsque la poutre est élevée à  $\frac{\pi}{2}$  rad, et négative lorsque l'angle d'élévation est supérieur à  $\frac{\pi}{2}$  rad, et positive lorsqu'il est inférieur à  $\frac{\pi}{2}$  rad, son expression mathématique est :

$$M_{v3} = -\Omega_h^2 \left\{ \left( \frac{m_t}{2} + m_{tr} + m_{ts} \right) l_t + \left( \frac{m_m}{2} + m_{mr} + m_{ms} \right) l_m + \left( \frac{m_b}{2} l_b + m_{cb} l_{cb} \right) \right\} \sin \alpha_v \cos \alpha_v \quad (\text{II.8})$$

Avec :

$$\Omega_h = \frac{d\alpha_h}{dt}$$

$\Omega_h$  : Vitesse angulaire de la poutre autour de l'axe vertical et  $\alpha_h$  est l'angle d'azimut de la poutre.

Ecriture sous forme compacte :

$$M_{v3} = -\Omega_h^2 (A + B + C) \sin \alpha_v \cos \alpha_v \quad (\text{II.9})$$

#### II.2.1.4 Moment de la force de friction $M_{v4}$

$$M_{v4} = -\Omega_v k_v \quad (\text{II.10})$$

Avec :

$$\Omega_v = \frac{d\alpha_v}{dt}$$

$\Omega_v$  : La vitesse angulaire autour de l'axe horizontal.

$k_v$  : Constante de friction à déterminer expérimentalement (par identification).

#### II.2.1.5 Moment d'inertie $J_v$

Le moment d'inertie par rapport à l'axe horizontal ne dépend pas de l'angle d'élévation car dans le plan vertical, la distribution de la masse est indépendante de l'angle  $\alpha_v$ .

$$\left\{ \begin{array}{l} J_{v1} = m_{mr} l_m^2 \\ J_{v2} = m_m \frac{l_m^2}{3} \\ J_{v3} = m_{cb} l_{cb}^2 \\ J_{v4} = m_b \frac{l_b^2}{3} \\ J_{v5} = m_{tr} l_t^2 \\ J_{v6} = m_t \frac{l_t^2}{3} \\ J_{v7} = \frac{m_{ms}}{2} r_{ms}^2 + m_{ms} l_m^2 \\ J_{v8} = m_{ts} r_{ts}^2 + m_{ts} l_t^2 \end{array} \right. \quad (\text{II.11})$$

Où :

$r_{ms}$  Le rayon de l'hélice principale.

$r_{ts}$  Le rayon de l'hélice secondaire.

## II.2.2 Modélisation du sous système d'azimut

Le mouvement dans le plan horizontal est repéré par l'angle d'azimut  $\alpha_h$ . On remarque, cependant, que la rotation de la poutre dans le plan horizontal est influencée par l'angle d'élévation  $\alpha_v$ .

### II.2.2.1 Influence de l'angle d'élévation sur le moment d'inertie

Le moment d'inertie d'un corps est fonction de la masse et de sa distribution géométrique dans ce corps. Pour cette raison, dans plusieurs systèmes mécaniques multidimensionnels comme c'est le cas du TRMS, on remarque que la distribution géométrique de la masse dépend de la position géométrique du corps, comme le montre la figure II.3.

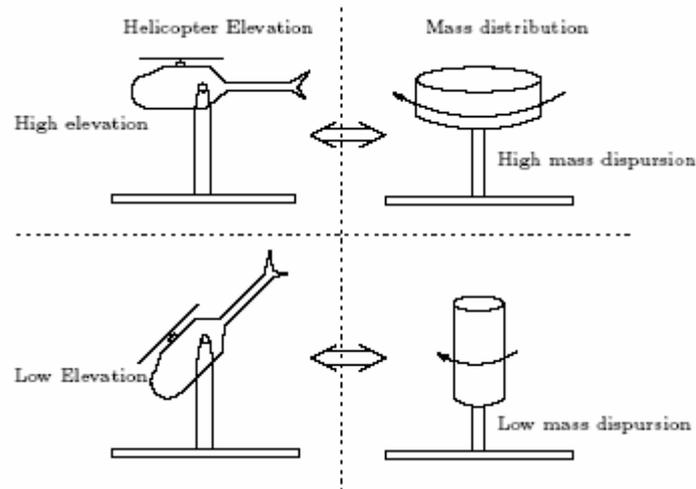


Figure II.3 : influence de l'angle d'élévation sur le moment d'inertie

### II.2.2.2 Influence de l'angle d'élévation sur le bras de levier des forces

Le moment d'une force dépend de la distance perpendiculaire entre cette force et l'axe de rotation. Dans notre cas, le mouvement dans le plan horizontal est généré par deux forces :

- La force de friction au niveau de l'articulation dont le bras de levier est supposé tellement petit qu'il n'est pas influencé par l'angle d'élévation ;
- Et la force aérodynamique qui possède un bras de levier variable en fonction de l'angle d'élévation  $\alpha_v$ .

De la même manière que pour le sous-système d'élévation, le mouvement de la poutre dans le plan horizontal peut être décrit par la deuxième loi de Newton :

$$M_h = J_h \frac{d^2 \alpha_h}{dt^2} \quad (\text{II.12})$$

Où :  $M_h$  est la somme des moments des forces agissantes dans le plan horizontal, et  $J_h$  est la somme des moments d'inertie par rapport à l'axe vertical.

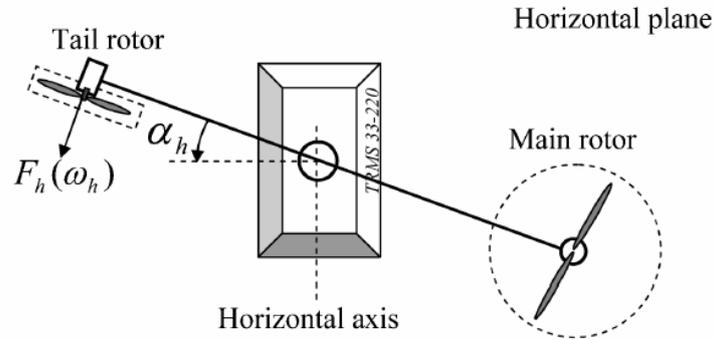
Ainsi :

$$M_h = \sum_{i=1}^2 M_{hi} \quad (\text{II.13})$$

$$J_h = \sum_{i=1}^8 J_{hi} \quad (\text{II.14})$$

### II.2.2.3 Moment de la force aérodynamique $M_{h1}$

Pour déterminer le moment de la force aérodynamique, considérons le cas présenté dans la figure II.4.



**Figure II.4 :** Moment de la force aérodynamique dans le plan horizontal

$$M_{h1} = l_t F_h(\omega_t) \cos \alpha_v \quad (\text{II.15})$$

Où :

$\omega_t$  La vitesse angulaire du rotor secondaire.

$F_h(\omega_t)$  Exprime la dépendance de la force aérodynamique de la vitesse angulaire du rotor secondaire.

### II.2.2.4 Moment de la force de friction $M_{h2}$

$$M_{h2} = -\Omega_h k_h \quad (\text{II.16})$$

Avec :  $k_h$  Constante de friction à déterminer expérimentalement par l'identification.

### II.2.2.5 Moment d'inertie $J_h$

Les moments d'inertie par rapport à l'axe vertical sont :

$$\left\{ \begin{array}{l} J_{h1} = \frac{m_m}{3} (l_m \cos \alpha_v)^2 \\ J_{h2} = \frac{m_t}{3} (l_t \cos \alpha_v)^2 \\ J_{h3} = \frac{m_b}{3} (l_b \sin \alpha_v)^2 \\ J_{h4} = m_{tr} (l_t \cos \alpha_v)^2 \\ J_{h5} = m_{mr} (l_m \cos \alpha_v)^2 \\ J_{h6} = m_{cb} (l_{cb} \sin \alpha_v)^2 \\ J_{h7} = \frac{m_{ts}}{2} r_{ts}^2 + m_{ts} (l_t \cos \alpha_v)^2 \\ J_{h8} = m_{ms} r_{ms}^2 + m_{ms} (l_m \cos \alpha_v)^2 \end{array} \right. \quad (\text{II.17})$$

Ou sous forme compacte :

$$J_h(\alpha_v) = D \cos^2 \alpha_v + E \sin^2 \alpha_v + F \quad (\text{II.18})$$

Avec :

$$\left\{ \begin{array}{l} D = \left( \frac{m_m}{3} + m_{mr} + m_{ms} \right) l_m^2 + \left( \frac{m_t}{3} + m_{tr} + m_{ts} \right) l_t^2 \\ E = \frac{m_b}{3} l_b^2 + m_{cb} l_{cb}^2 \\ F = m_{ms} r_{ms}^2 + \frac{m_{ts}}{2} r_{ts}^2 \end{array} \right.$$

Où :

$J_{tr}$  Le moment d'inertie dans le moteur secondaire.

$J_{mr}$  Le moment d'inertie dans le moteur principal.

### II.2.3 Modélisation des propulseurs

Le TRMS possède deux propulseurs constitués chacun d'un moteur à courant continu plus une hélice, un pour le rotor principal et l'autre pour le rotor de queue (rotor secondaire). Les deux moteurs DC sont identiques, mais avec des charges mécaniques différentes.

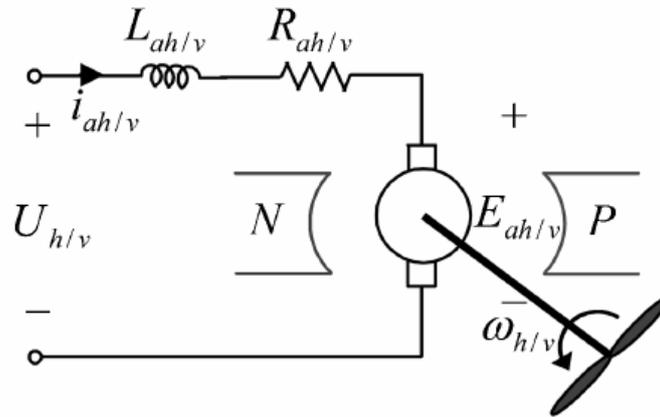


Figure II.5 : Circuit de l'un des propulseurs du TRMS

L'équation différentielle du moteur DC est supposée du premier ordre, en négligeant la constante de temps électrique devant la constante de temps mécanique.

Cette équation différentielle est donnée par :

$$I\dot{\omega} = \frac{K_i}{R}(u - K_b\omega) - T_L \quad (\text{II.19})$$

Avec :

$\omega$  : Vitesse angulaire du moteur (rad/s).

$u$  : Tension de commande (V).

$I$  : Moment d'inertie.

$R$  : Résistance de l'armature.

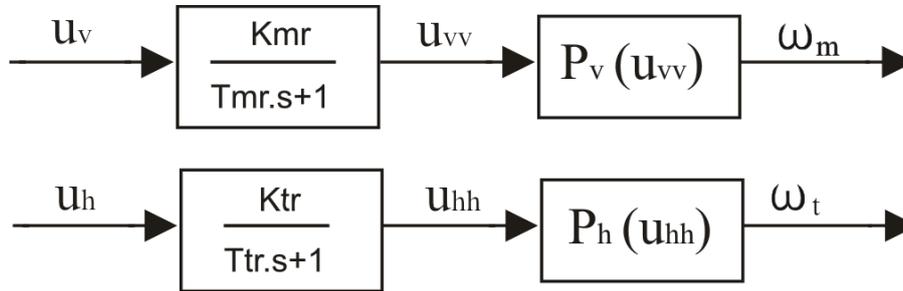
$K_b$  : Constante de la FEM.

$K_i$  : Constante du couple.

$T_L$  : Couple résistant généré par la charge.

La charge  $T_L$ , représente le couple résistant généré par les frottements mécaniques et les frottements aérodynamiques de l'hélice avec l'air. Ces frottements sont difficilement modélisables. Pour cela, leurs caractéristiques statiques sont déterminées expérimentalement. Elles dépendent de la forme et des dimensions de chaque hélice. Dans le TRMS, l'hélice du rotor principal est différente de celle du rotor secondaire.

Pour cela, on a introduit deux nouvelles variables :  $U_{vv}$  pour le vertical et  $U_{hh}$  pour l'horizontal, comme l'illustre la figure (II.6).



**Figure II.6 :** schéma bloc des propulseurs

Où :

$T_{mr}$  : La constante du temps du moteur principal.

$T_{tr}$  : La constante du temps du moteur secondaire.

$K_{mr}$  : Le gain statique du moteur principal.

$K_{tr}$  : Le gain statique du moteur secondaire.

$P_v(U_{vv})$  et  $P_h(U_{hh})$  sont les caractéristiques statiques des frottements, déterminées expérimentalement.

Ainsi, les sous-systèmes propulseurs sont décrits par des équations différentielles du premier ordre avec des fonctions non linéaires à la sortie :

$$\begin{cases} \frac{du_{vv}}{dt} = \frac{1}{T_{mr}}(-u_{vv} + K_{mr}u_v) \\ \omega_m = p_v(u_{vv}) \end{cases} \quad (\text{II.20})$$

$$\begin{cases} \frac{du_{hh}}{dt} = \frac{1}{T_{tr}}(-u_{hh} + K_{tr}u_h) \\ \omega_t = p_h(u_{hh}) \end{cases} \quad (\text{II.21})$$

#### II.2.4 Effet gyroscopique

Lorsqu'on met une hélice en rotation autour d'un axe, plusieurs phénomènes surviennent, notamment, l'effet gyroscopique ou le mouvement de précession.

De plus, un objet en rotation possède un moment cinétique (ou moment angulaire) proportionnel à son inertie et sa vitesse angulaire :  $\vec{S} = I \cdot \vec{\omega}$ . Le théorème de Koeing met en relation le moment cinétique et la somme des moments :

$$\frac{d\vec{S}}{dt} = \sum_i \vec{M}(\vec{F}_i)$$

En effet, dans un hélicoptère on a une hélice (masse) qui tourne autour d'un axe principal tout comme dans un gyroscope. Ce que l'on retient en premier lieu, c'est que l'axe et l'hélice vont toujours tendre à être sur le même axe de rotation, et lorsqu'une force viendra perturber le système (propulsion, frottement de l'air sur les pales,...), celui-ci voudra la compenser en entamant une rotation autour d'un axe du plan horizontal parallèle à la force appliquée.

Donc par analogie, toute force appliquée ailleurs que sur le centre de masse du système provoquera un mouvement résultant perpendiculaire, en plus de la résistance de l'air lors du mouvement du simulateur.

D'où les relations suivantes :

$$S_v = J_v \Omega_v - J_{tr} \omega_t \quad (\text{II.22})$$

$J_{tr} \omega_t$  : est le moment cinétique généré par le rotor de queue pour contrer l'effet gyroscopique dû à la rotation de l'hélice principale.

$$S_h = J_h(\alpha_v) \Omega_h - J_{mr} \omega_m \cos \alpha_v \quad (\text{II.23})$$

$J_{mr} \omega_m \cos \alpha_v$  : est le moment cinétique généré par le rotor principal pour contrer l'effet gyroscopique dû à la rotation de l'hélice du rotor secondaire.

### II.3 Modèle d'état non linéaire

En utilisant les relations mathématiques développées précédemment, on peut réécrire les équations régissant le mouvement du système comme suit :

$$\frac{dS_v}{dt} = l_m F_v(\omega_m) - \Omega_v k_v + g((A - B) \cos \alpha_v - C \sin \alpha_v) - \frac{1}{2} \Omega_h^2 (A + B + C) \sin 2\alpha_v \quad (\text{II.24})$$

$$\frac{d\alpha_v}{dt} = \Omega_v \quad (\text{II.25})$$

$$\Omega_v = \frac{S_v + J_{tr} \omega_t}{J_v} \quad (\text{II.26})$$

$$\frac{du_{vv}}{dt} = \frac{1}{T_{mr}} (-u_{vv} + K_{mr} u_v) \quad (\text{II.27})$$

$$\omega_m = P_v(u_{vv})$$

$$\frac{dS_h}{dt} = l_t F_h(\omega_t) \cos \alpha_v - \Omega_h k_h \quad (\text{II.28})$$

$$\frac{d\alpha_h}{dt} = \Omega_h \quad (\text{II.29})$$

$$\Omega_h = \frac{S_h + J_{mr} \omega_m \cos \alpha_v}{J_h(\alpha_v)} \quad (\text{II.30})$$

$$\frac{du_{hh}}{dt} = \frac{1}{T_{tr}} (-u_{hh} + K_{tr} u_h) \quad (\text{II.31})$$

$$\omega_t = P_h(u_{hh})$$

Ainsi, en classifiant les différentes variables comme suit :

- Vecteur d'entrée :  $U = [U_v \ U_h]^T$
- Vecteur d'état :  $X = [\alpha_v \ S_v \ u_{vv} \ \alpha_h \ S_h \ u_{hh}]^T$
- Vecteur de sortie :  $Y = [\alpha_v \ \alpha_h]^T$

On obtient une représentation d'état non linéaire sous la forme : 
$$\begin{cases} \dot{X} = f(X) + g(X)U \\ Y = h(X) \end{cases}$$

où il y a 6 variables d'état.

Représentation d'état :

$$\left\{ \begin{aligned} \dot{x}_1 &= \frac{x_2 + J_{tr}P_h(x_6)}{J_v} \\ \dot{x}_2 &= l_m F_v(P_v(x_3)) - k_v \left[ \frac{x_2 + J_{tr}P_h(x_6)}{J_v} \right] + g((A - B)\cos(x_1) - C \sin(x_1)) \\ &\quad - \frac{1}{2} \left[ \frac{x_5 + J_{mr}P_v(x_3)\cos(x_1)}{J_h(x_1)} \right]^2 (A + B + C)\sin(2x_1) \\ \dot{x}_3 &= \frac{1}{T_{mr}}(-x_3 + K_{mr}u_1) \\ \dot{x}_4 &= \frac{x_5 + J_{mr}P_v(x_3)\cos(x_1)}{J_h(x_1)} \\ \dot{x}_5 &= l_t F_h(P_h(x_6))\cos(x_1) - k_h \left[ \frac{x_5 + J_{mr}P_v(x_3)\cos(x_1)}{J_h(x_1)} \right] \\ \dot{x}_6 &= \frac{1}{T_{tr}}(-x_6 + K_{tr}u_2) \end{aligned} \right. \quad (\text{II.32})$$

La figure (II.7), montre le diagramme fonctionnel du TRMS :

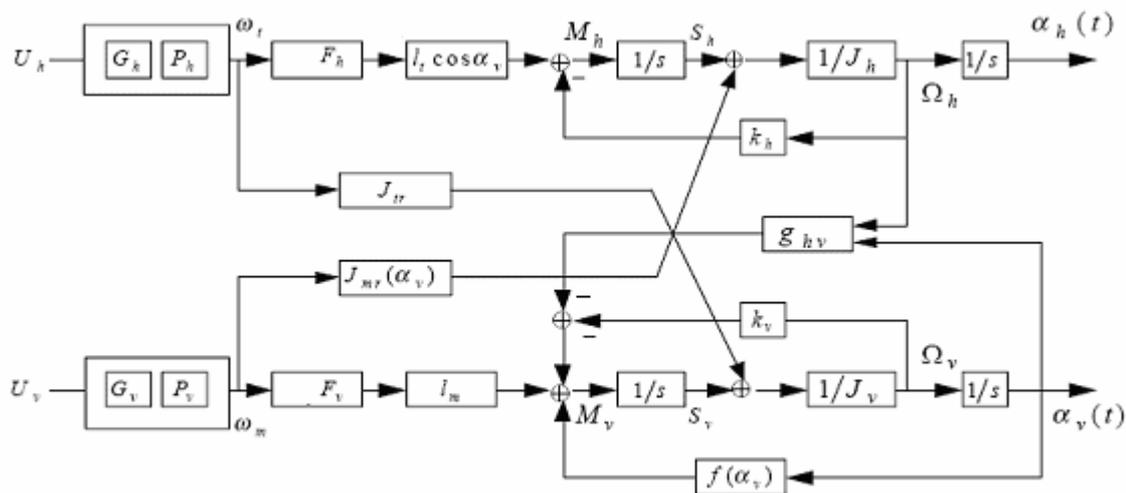


Figure II.7: Schéma bloc du TRMS

## II.4 Le modèle découplé

En contraignant le mouvement de la tige dans un seul plan soit l'horizontal ou le vertical, on obtient deux sous modèles ayant chacun un degré de liberté.

### II.4.1 Modèle 1 DDL vertical

Ce modèle est dérivé du modèle couplé, en fixant l'angle d'azimut  $\alpha_h$ , et en posant  $u_h = 0$ .

On choisit le vecteur d'état suivant :

$$X = [\alpha_v \ S_v \ u_{vv}]^T$$

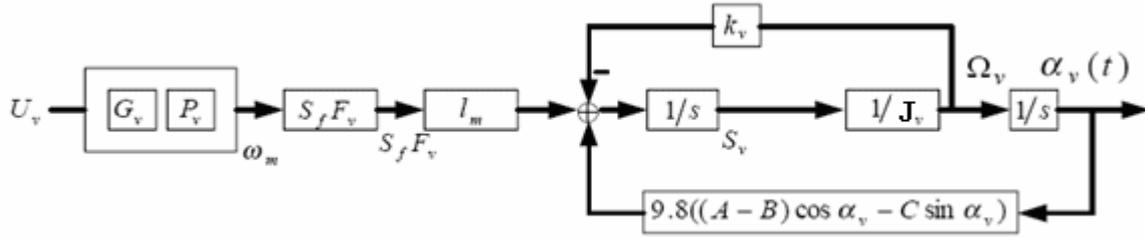


Figure II.8 : Schéma bloc du modèle vertical

La représentation d'état est alors :

$$\begin{cases} \dot{x}_1 = \frac{x_2}{J_v} \\ \dot{x}_2 = l_m F_v(P_v(x_3)) - k_v \cdot \frac{x_2}{J_v} + g((A-B)\cos(x_1) - C\sin(x_1)) \\ \dot{x}_3 = \frac{1}{T_{mr}} (-x_3 + K_{mr} u_1) \end{cases} \quad (II.33)$$

**Remarque :**

Le modèle vertical obtenu ne dépend pas de l'angle d'azimut  $\alpha_h$ .

### II.4.2 Modèle 1 DDL horizontal

De la même façon que pour le modèle vertical, dans le modèle couplé on pose  $\alpha_v = \alpha_{v0}$  et  $u_v = 0$ . Et on choisit  $X = [\alpha_h \ S_h \ u_{hh}]^T$  comme vecteur d'état.

Le modèle horizontal est ainsi :

$$\begin{cases} \dot{x}_1 = \frac{1}{J_h(\alpha_{v0})} x_2 \\ \dot{x}_2 = l_t F_h(P_h(x_3)) \cos(\alpha_{v0}) - \frac{k_h}{J_h(\alpha_{v0})} x_2 \\ \dot{x}_3 = \frac{1}{T_{tr}} (-x_3 + K_{tr} u_h) \end{cases} \quad (\text{II.34})$$

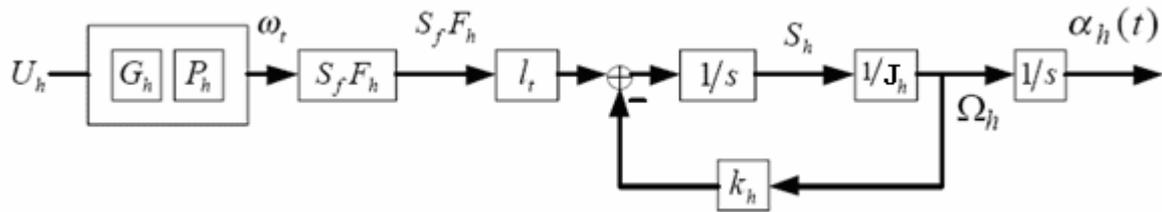


Figure II.9 : Schéma bloc du modèle horizontal

## II.5 Simulations

Afin de pouvoir utiliser le modèle du TRMS pour des simulations ou pour développer des lois de commandes, nous devons déterminer les différents paramètres et constantes de temps du système, ainsi que les caractéristiques statiques des fonctions non linéaires en utilisant l'identification.

Les courbes obtenues expérimentalement (annexe B), sont approximées par des polynômes en utilisant la fonction *polyfit* de Matlab.

Ainsi, pour le rotor principal on obtient les expressions suivantes:

**Fonction non linéaire du rotor principal :**

$$P_v(U_{vv}) = 90,99 \cdot U_{vv}^6 + 599,73 \cdot U_{vv}^5 - 129,26 \cdot U_{vv}^4 - 1238,64 \cdot U_{vv}^3 + 63,45 \cdot U_{vv}^2 + 1283,41 \cdot U_{vv} \quad (\text{II.36})$$

**Fonction aérodynamique principale :**

$$F_v(\omega_m) = -3,48 \cdot 10^{-12} \cdot \omega_m^5 + 1,09 \cdot 10^{-9} \cdot \omega_m^4 + 4,123 \cdot 10^{-6} \cdot \omega_m^3 - 1,632 \cdot 10^{-4} \cdot \omega_m^2 + 9,544 \cdot 10^{-2} \cdot \omega_m \quad (\text{II.37})$$

Et pour le rotor secondaire on a :

**Fonction non linéaire du rotor secondaire :**

$$P_h(U_{hh}) = 2020 \cdot U_{hh}^5 - 194,69 \cdot U_{hh}^4 - 4283,15 \cdot U_{hh}^3 + 262,27 \cdot U_{hh}^2 + 3796,83 \cdot U_{hh} \quad (\text{II.38})$$

**Fonction aérodynamique secondaire :**

$$F_t(\omega_t) = -3 \cdot 10^{-14} \cdot \omega_t^5 - 1.595 \cdot 10^{-11} \cdot \omega_t^4 + 2.511 \cdot 10^{-7} \cdot \omega_t^3 - 1.808 \cdot 10^{-4} \cdot \omega_t^2 + 0.0801 \cdot \omega_t \quad (\text{II.39})$$

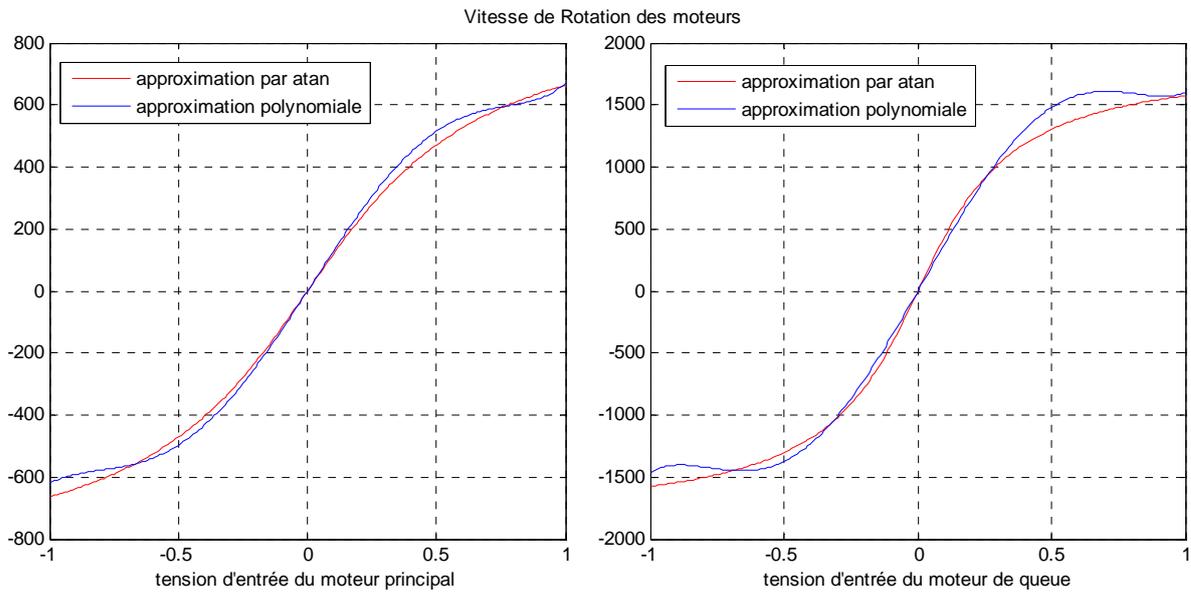
Pour les rotors principal et secondaire, les polynômes décrivant les fonctions non linéaires des moteurs peuvent être remplacés par :

Rotor principal:

$$P_v(U_{vv}) = 600 \cdot \text{atan}(2 \cdot U_{vv}) \quad (\text{II.40})$$

Rotor secondaire:

$$P_h(U_{hh}) = 1200 \cdot \text{atan}(3.8 \cdot U_{hh}) \quad (\text{II.41})$$



**Figure II.10 :** approximation par des atan des caractéristiques non linéaires des moteurs (moteur principal (gauche), moteur de queue (droite))

Les paramètres du modèle sont représentés sous forme de tableau, ils seront utilisés lors de la simulation du modèle [6]

**Tableau II.1 :** Les paramètres du modèle

Paramètre	Valeur
A	0.0947
B	0.1105
C	0.011702
D	0.048814

E	0.0016087
F	0.006225
$J_v$ (kg.m <sup>2</sup> )	0.055448
$J_{tr}$ (kg.m <sup>2</sup> )	1.6543 10 <sup>-5</sup>
$J_{mr}$ (kg.m <sup>2</sup> )	2.65 10 <sup>-5</sup>
$l_m$ (m)	0.24
$l_t$ (m)	0.25
$T_{mr}$	1.432
$T_{tr}$	0.3842
$K_{mr}$	1
$K_{tr}$	1
$K_v$	0.00545371
$K_h$	0.0095

Les figures II.11 et II.12, montrent les réponses libres et forcées du simulateur d'hélicoptère, avec les conditions initiales :  $\alpha_v = \alpha_h = 0^\circ$  :

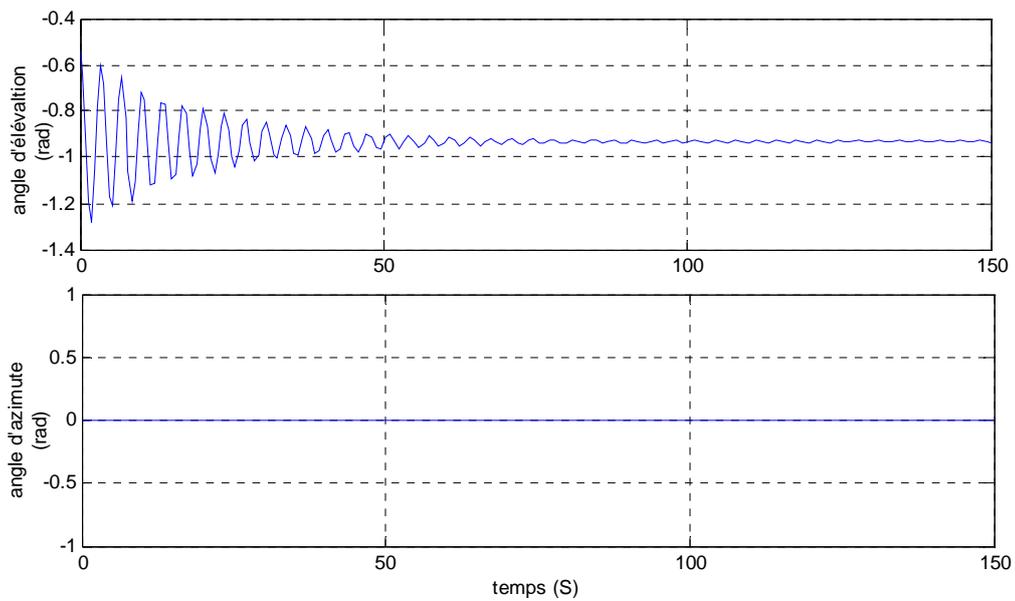
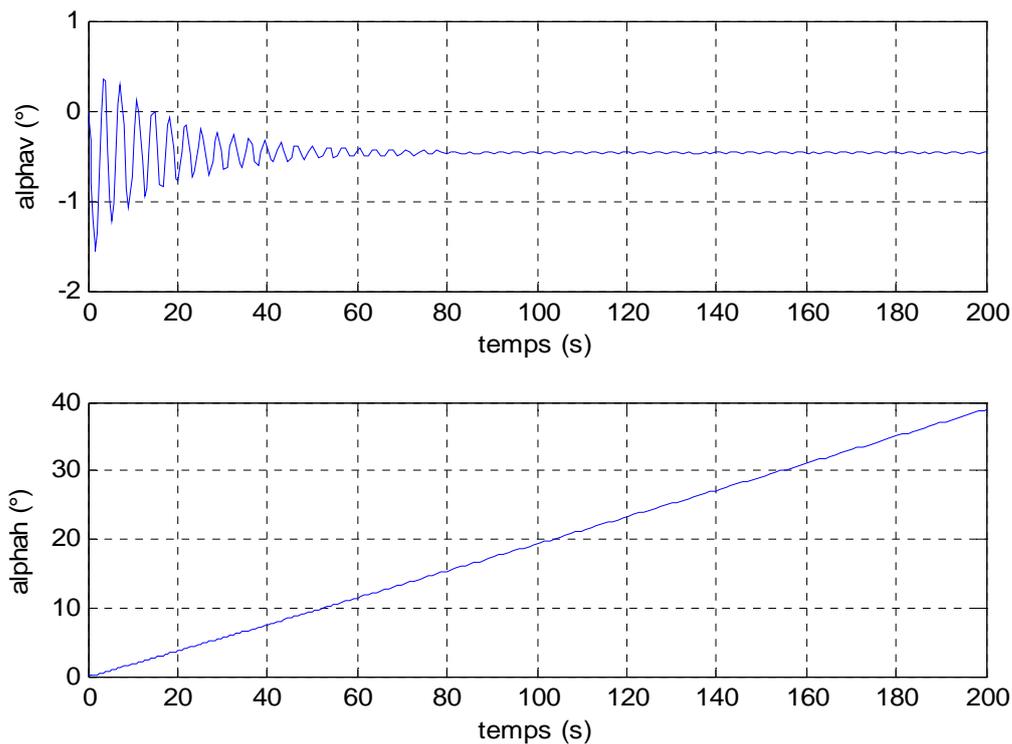


Figure II.11 : réponses libres du TRMS avec  $\alpha_{v0} = \alpha_{h0} = 0^\circ$  °



**Figure II.12 :** réponses du TRMS pour  $U_v=U_h=0.5$  et  $\alpha_{v0} = \alpha_{h0} = 0^\circ$

La réponse libre du sous-système d'élévation est oscillatoire amortie, cela est due aux forces gravitationnelles qui agissent uniquement sur le plan vertical, et poussent le simulateur à se stabiliser en un point d'équilibre  $\alpha_v = -0.93 \text{ rad}$ . Par contre, le sous-système d'élévation reste dans sa position d'origine tant qu'il n'y a pas d'excitation du Rotor de queue. Lorsqu'on excite le système, le sous-système vertical tend vers un nouveau point d'équilibre car la commande 0.5 n'est pas suffisante pour l'élever vers un angle supérieur. Cependant, le sous système horizontal a le comportement d'un intégrateur, il diverge même pour de petites excitations. Ceci est dû essentiellement à la faible inertie du mouvement horizontal où il n'y a pas de force de gravité qui s'oppose au mouvement.

## II.6 Conclusion

Dans ce chapitre, on a établi un modèle mathématique du simulateur d'hélicoptère TRMS, celle-ci a abouti à une représentation d'état fortement non linéaire et couplée. Cette modélisation nous permettra par la suite de concevoir des contrôleurs.

## Chapitre III

### *Quelques aspects de la commande multi contrôleur*

#### III.1 Introduction

Les systèmes non linéaires, ayant d'importantes interactions entre les différents états, présentent un modèle mathématique difficile à obtenir et souvent incertain. Toutefois, il est possible d'avoir plusieurs modèles assez simples du même procédé associés à différents environnements. Ainsi, le problème de contrôle se simplifierait considérablement si l'on arrive à synthétiser un contrôleur performant à chaque modèle local. De ce fait, on peut envisager d'approximer le système non linéaire par un certain nombre de modèles linéaires valides au voisinage de leurs points de fonctionnement, ensuite, synthétiser des contrôleurs locaux via la théorie des systèmes linéaires. [7]

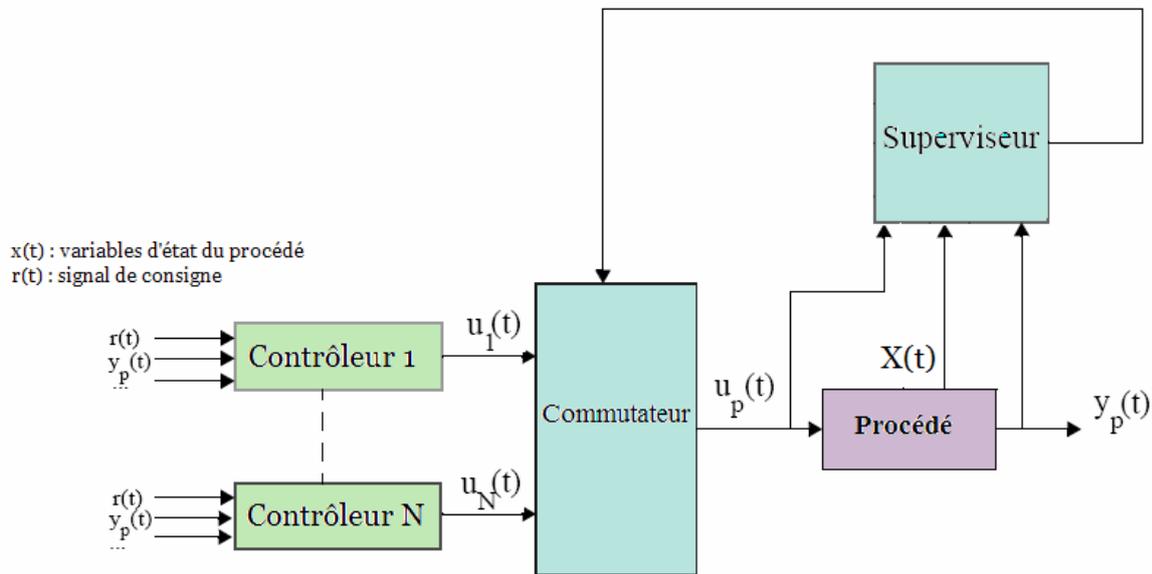
#### III.2 Concept général de la commande multi contrôleurs

La synthèse de la commande multi contrôleurs s'effectue en deux étapes :

- La première consiste à concevoir un ensemble fini de  $N$  contrôleurs locaux relatifs à leurs environnements respectifs, ces contrôleurs peuvent être de structure et de natures différentes.
- La seconde étape se résume à constituer le signal de commande finale qui sera appliqué au système à partir des sorties des différents contrôleurs locaux. Cette tâche est assurée par le « superviseur », qui, à partir des différentes informations du système (commande, sortie du procédé, variables internes du système), donne l'ordre au commutateur de basculer vers le ou les contrôleurs les plus aptes à garantir les performances à chaque instant.

Dans le cas où les correcteurs de la commande multi contrôleurs sont de même structure, cette dernière peut être assimilée à une commande adaptative dont les paramètres

varient au cours du temps de façon discontinue au moment des commutations. Une analogie avec la commande floue de type Takagi-Sugeno est également possible. Celle-ci propose également une partition de l'univers du discours. Les conclusions sur le choix et l'activation seront transmises au commutateur qui procédera à la pondération des sorties des différents contrôleurs pour former le signal de commande appliqué au système.



**Figure III.1 :** schéma de principe de la commande multi contrôleurs (classe indirecte)

On peut classer la commande multi contrôleur en deux catégories :

- *commande indirecte* : cette approche contient un superviseur (figure III.1) capable de déterminer l'environnement où évolue le système. Il transmet des informations au commutateur par le biais desquels il choisira un ou plusieurs contrôleurs. Cette commutation est basée sur la prédiction.
- *Commande directe* : cette classe ne dispose pas de superviseur (figure III.2). Le commutateur choisit lui-même les correcteurs locaux les plus performants à partir d'informations mesurées.

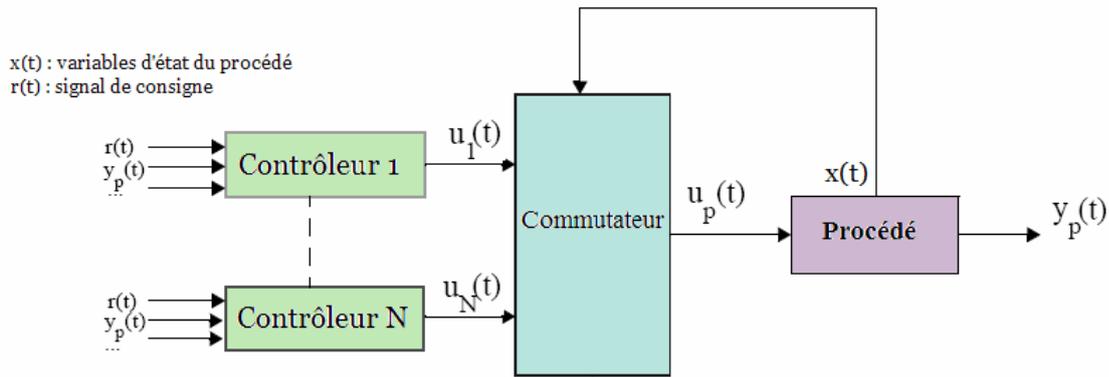


Figure III.2 : schéma de principe de la commande multi contrôleurs (classe directe)

### III.3 Commande multi contrôleurs et multi modèle

Dans la littérature, le terme commande multi modèle a été attribué à une commande suffisamment robuste pour garantir les performances voulues en boucle fermée d'un procédé présentant des incertitudes paramétriques. Ainsi, à partir de différents modèles caractérisant chacun un environnement bien particulier, un unique contrôleur est synthétisé de manière à donner de bonnes performances (stabilité robuste, rejet de perturbations, calibrage des fonctions de sensibilité) pour tous les modèles. La principale différence avec l'approche multi contrôleurs réside en l'absence de commutation ou de supervision. En effet, un jeu de paramètres d'un unique contrôleur est calculé afin de satisfaire tous les modèles. [7]

### III.4 Commutation «franche»

#### III.4.1 approche directe

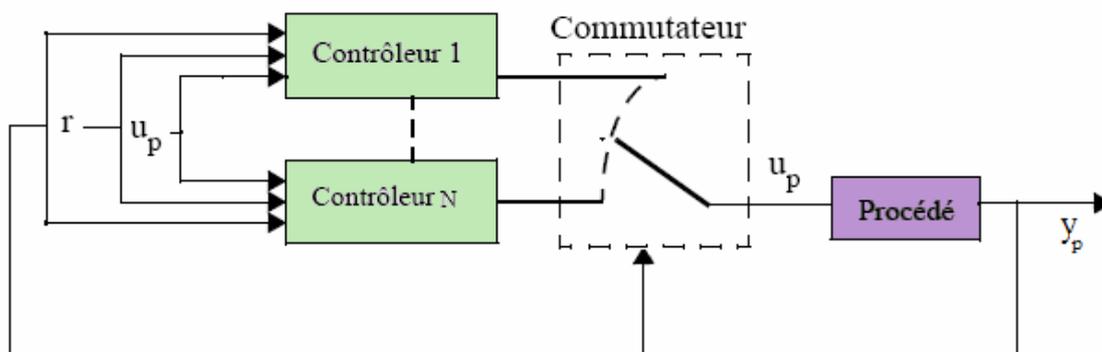


Figure III.3 : commande multi contrôleurs approche directe avec décision suivant la sortie du système

Dans cette partie, la commutation concerne la sélection d'un seul contrôleur local à la fois.

La commutation sera ainsi qualifiée de «franche». Pour cette classe, certains ouvrages (par exemple [2] ou encore [9]) qualifient ce type de commande d'« hybride » dans le sens où cette commande prend des valeurs continues et discrètes au sein d'une structure commune.

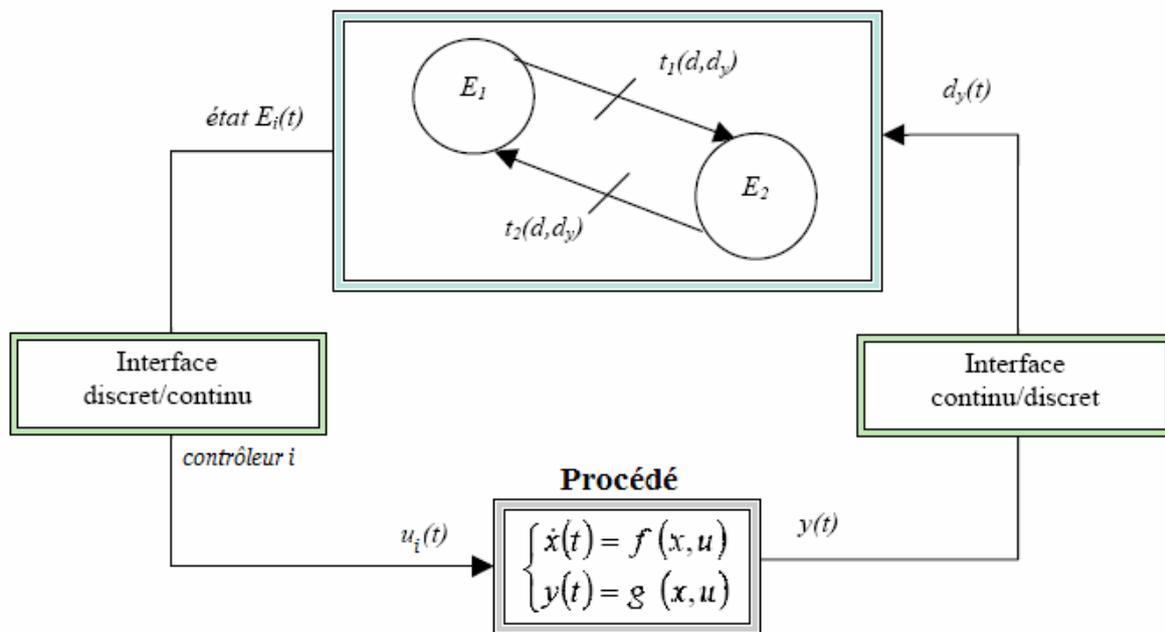


Figure III.4 : schéma de principe d'une commutation régie par un système à événement discret

La commutation des contrôleurs s'effectue donc selon une séquence prédéfinie programmée dans un système à événement discret (SED). Le choix des états et des règles de transitions est basé sur la sortie du procédé (figure III.4). Dans le cas où le système est linéarisé autour de plusieurs points de fonctionnement, les transitions d'un événement à un autre (et donc d'un contrôleur à l'autre) se produit lors du passage d'un environnement (voisinage de point de fonctionnement) à un autre.

### III.4.1 approche indirecte

Dans cette classe le superviseur est constitué généralement d'un ensemble de modèles ou de prédicateurs capables d'identifier l'environnement dans lequel évolue le procédé. Cette information préalablement traitée est envoyée au commutateur (voir annexe B).

### III.5 Gestion des Commutation

La spécification des performances d'une commande multi contrôleurs est peu abordée dans la littérature, mais dépend essentiellement des performances des contrôleurs locaux. La stabilité peut être étudiée grâce aux fonctions de Lyapunov continue et discontinues. Néanmoins, les commutations induisent des sauts dans la réponse du système auxquels il faut pallier.

Dans la majorité des cas, la commande hybride (multi-contrôleurs à commutation franche) a été utilisée dans les systèmes électroniques (cas d'une commande PWM, bang-bang). Les commutations induisent alors des sauts dans la réponse du système. Ces sauts, dans les systèmes électroniques, sont désirés et leurs effets à long terme ne produisent qu'un vieillissement des composants qui est pris en compte lors de leur fabrication. Cependant, dans certains systèmes, comme un hélicoptère, les sauts dans la réponse induits par les commutations sont très souvent néfastes et s'ils ne sont pas limités peuvent induire la perte de contrôle qui est impardonnable pour un hélicoptère.

De ce fait, sans pouvoir déterminer les performances de la commande multi contrôleurs, il est nécessaire de trouver des solutions pour atténuer ou supprimer les sauts induits lors des commutations.

#### III.5.1 Réinitialisation de la commande

La première idée pour gérer les sauts dans la réponse du système est bien entendu d'annuler ces sauts directement dans la commande lors d'une commutation. Ainsi, il faut pourvoir au moins s'assurer que la commande soit une fonction continue du temps, car sa discontinuité à la commutation excite transitoirement le système provoquant ainsi des sauts.

Explicitons cette solution par un exemple simple :

Soit le système du premier ordre décrit par la fonction de transfert :

$$G(s) = \frac{1}{\tau \cdot s + 1} \quad (\text{III.1})$$

Pour l'exemple on pose  $\tau = 5$ .

Deux contrôleurs PI sont alors synthétisés avec des dynamiques différentes :

$$C^i(s) = \frac{K_I^i}{s} + K_P^i \quad (III.2)$$

Où  $i$  représente le numéro du contrôleur. Ils sont sélectionnés par une entrée discrète exogène.

Le schéma de commande est alors celui d'une rétroaction classique

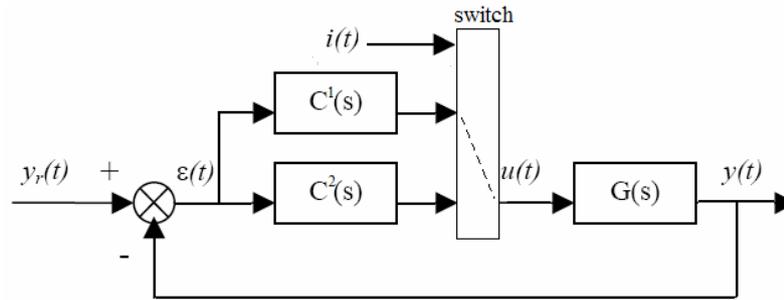


Figure III.5 : schéma de commande avec entrée de sélection discrète exogène.

Pour l'exemple on prend :

$$\begin{cases} K_P^1 = 5 \\ K_I^1 = 1 \end{cases} \quad \begin{cases} K_P^2 = 20 \\ K_I^2 = 4 \end{cases}$$

Supposons que, sur l'intervalle de temps  $[0 ; t_0[$ , le mode est  $i = 1$  ; le premier contrôleur est actif. La commande est alors:

$$u(t) = K_P^1 \cdot \varepsilon(t) + K_I^1 \cdot \int_0^t \varepsilon(\tau) \cdot d\tau, \quad t \in [0; t_0[ \quad (III.3)$$

Avec : 
$$\varepsilon(t) = y_r(t) - y(t), \quad \forall t \quad (III.4)$$

A l'instant  $t_0$  (pour l'exemple  $t_0=15$  s) la commande commute et le contrôleur 2 prends le relais, la commande  $u$  devient alors pour l'intervalle]  $t_0 ; \infty$  [égale a:

$$u(t) = K_P^2 \cdot \varepsilon(t) + K_I^2 \cdot \int_0^t \varepsilon(\tau) \cdot d\tau, \quad t \in ]t_0; \infty[ \quad (III.5)$$

La commande peut donc subir une discontinuité qui peut être exprimée en prenant respectivement les limites à gauche et à droite de l'instant de commutation, soit :

$$\begin{aligned}
u(t_0^+) - u(t_0^-) &= K_P^2 (y_r(t_0^+) - y(t_0^+)) - K_P^1 (y_r(t_0^-) - y(t_0^-)) \\
&+ K_I^2 \int_0^{t_0^+} (y_r(\tau) - y(\tau)) \cdot d\tau - K_I^1 \int_0^{t_0^-} (y_r(\tau) - y(\tau)) \cdot d\tau
\end{aligned} \tag{III.6}$$

La sortie du système étant une fonction continue (mais non dérivable en cas de discontinuité sur la commande) et en supposant également que la consigne est une fonction continue lors des instants de commutations, on peut simplifier l'équation (III.10) comme suit :

$$u(t_0^+) - u(t_0^-) = (K_P^2 - K_P^1) \cdot \varepsilon(t_0) + (K_I^2 - K_I^1) \int_0^{t_0} \varepsilon(\tau) \cdot d\tau \tag{III.7}$$

La discontinuité de commande va se traduire par une discontinuité sur la dérivée du signal de sortie qui provoquera un saut sur la réponse du système.

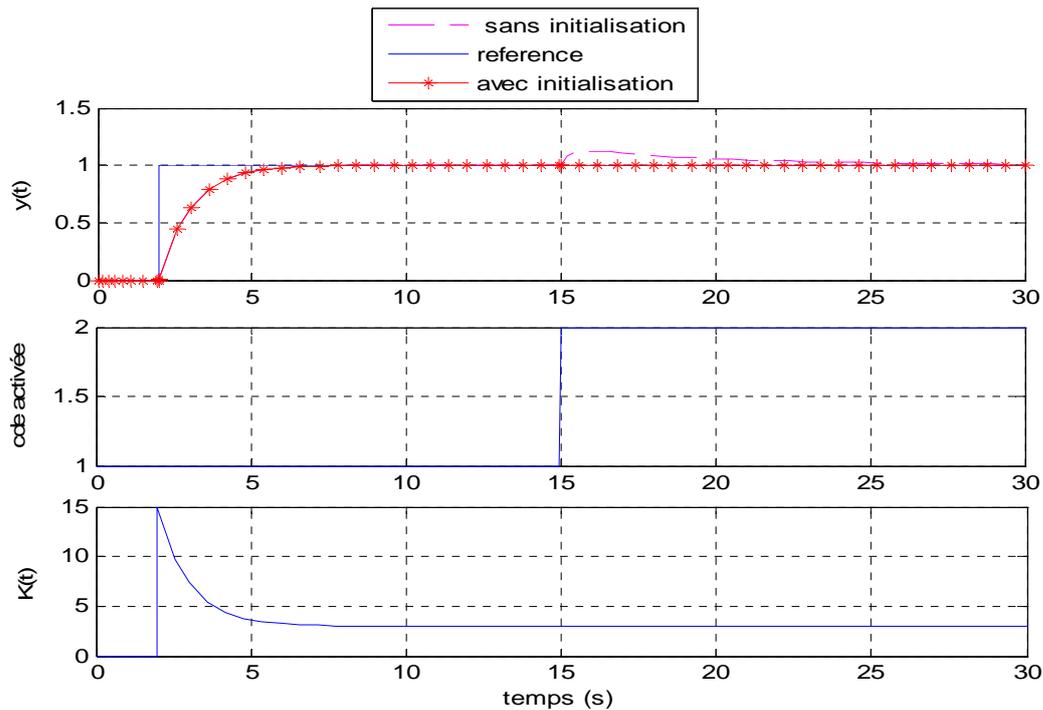
Mais, grâce à la relation (III.11), il est possible de calculer en permanence la quantité :

$$K(t) = (K_P^2 - K_P^1) \cdot \varepsilon(t) + (K_I^2 - K_I^1) \int_0^t \varepsilon(\tau) \cdot d\tau \tag{III.8}$$

Et en appliquant la commande réinitialisée après la commutation :

$$u_{reinit}(t) = \begin{cases} u(t) & t < t_0 \\ u(t) - K(t_0) & t \geq t_0 \end{cases} \tag{III.9}$$

Cette nouvelle commande ne subit plus de discontinuité et permet ainsi de ne pas induire de saut dans la réponse du système comme le montre les simulations suivantes :



**Figure III.6 :** Evolutions de la consigne (---), de la sortie (—) sans réinitialisation, de la sortie (-\*) avec réinitialisation, du mode fonctionnement et de la constante de réinitialisation de la commande multi-contrôleurs réinitialisée.

La réinitialisation permet ainsi d'annuler les sauts éventuels lors des commutations de la commande. Néanmoins, cette méthode présente des inconvénients :

- il faut nécessairement calculer en ligne la constante de réinitialisation puisque l'instant de commutation est inconnu ;
- la difficulté de calcul analytique croît avec l'augmentation de l'ordre du modèle du procédé et des contrôleurs commutés.

Dans le cas général, il est fastidieux, voire analytiquement impossible, d'écrire les conditions de continuité permettant de réinitialiser la commande. Il existe, cependant, une deuxième approche plus simple sur le plan analytique consistant en la poursuite des commandes commutées.

### III.5.2 Poursuite de la commande

La deuxième approche possible ne consiste plus à annuler les sauts dans la réponse du système, mais plutôt à les atténuer grâce à un filtrage des commandes commutées. Une

architecture de commande générique est proposée où un contrôleur actif  $C_A$  et un contrôleur latent  $C_L$  commutent dans l'asservissement du procédé  $G$ . [Graebe93]

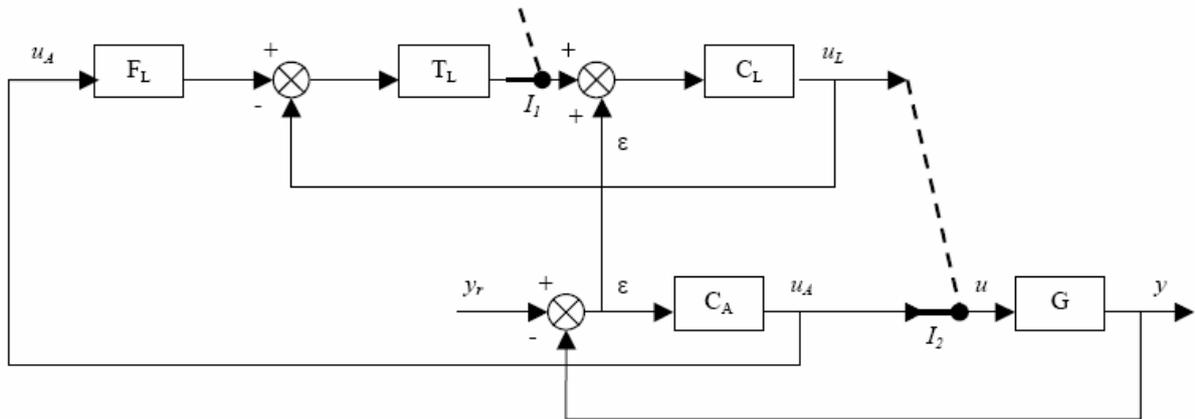


Figure III.7 : Architecture du bouclage contenant la poursuite de commande

L'architecture se décompose en deux boucles :

- le contrôleur actif assure le suivi de consigne par une première boucle de commande ;
- le contrôleur latent suit comme consigne la commande active en dépit de l'erreur de la première boucle qui agit alors comme perturbation exogène.

A la commutation, les deux interrupteurs  $I_1$  et  $I_2$  commutent, la commande latente devient alors la commande active, et la commande active cesse d'être utilisée : l'architecture de commande s'inverse et les deux régulateurs échangent leurs rôles préparant ainsi la commutation suivante.

Avant que le contrôleur  $C_L$  soit activé, la deuxième boucle d'asservissement peut être extraite sous la forme donnée par la figure suivante :

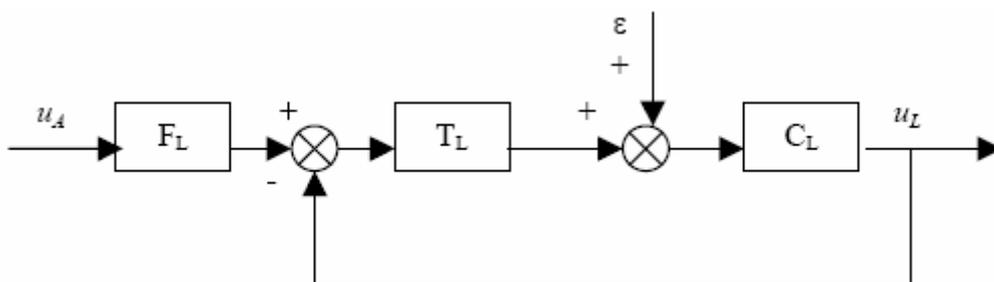


Figure III.8 : Architecture de commande pour le suivi de la commande active par la commande latente

La commande  $u_L$  est asservie à  $u_A$  et le contrôleur latent  $C_L$  joue alors le rôle de procédé dans cet asservissement. Les contrôleurs  $F_L$  et  $T_L$  sont alors respectivement les contrôleurs par anticipation et par rétroaction à concevoir; les règles classiques de synthèse s'appliquent alors. La commande latente est en permanence asservie à la commande active avant la commutation. Lors de la commutation, les discontinuités sur la commande sont minimisées ce qui limite par conséquent les sauts dans la réponse du système.

Cette méthode permet donc de réduire notablement les sauts lors des commutations de la commande multi contrôleurs. La complexité de sa mise en oeuvre est seulement liée aux contrôleurs utilisés dans la boucle d'asservissement de la commande active et non directement au procédé lui-même (contrairement à la première approche). Les règles classiques de synthèse d'une loi de commande par anticipation et par rétroaction peuvent être utilisées pour l'asservissement à commande active.

Deux inconvénients subsistent néanmoins :

- la commande hybride nécessite d'être augmentée par l'architecture proposée en Figure III.7;
- les sauts dans la commande ne sont pas explicitement annulés (contrairement à la première approche) mais diminués par un asservissement efficace de la commande active.

La dernière approche à envisager pour la gestion des commutations ne s'appuie plus sur un traitement des commandes continues locales mais sur le commutateur lui même. Au lieu de raisonner sur celui-ci avec un SED qui ne comporte que des états disjoints, une logique ensembliste est préférée en s'appuyant sur la logique floue.

### **III.6 Commutation «floue» et approche directe**

Dans le cadre de cette méthode, plusieurs contrôleurs locaux peuvent entrer en jeu pour élaborer le signal de commande final. La notion de fonctions de validité est remplacée par des fonctions d'appartenance issues du formalisme flou qui se trouve à la base d'un nouveau type de commutation que nous appelons «commutation floue». Ce terme peut paraître ambiguë car plusieurs chercheurs associent la commutation à un choix d'un seul régulateur parmi plusieurs. Dans notre cas, nous avons étendu cette définition à la sélection de plusieurs.

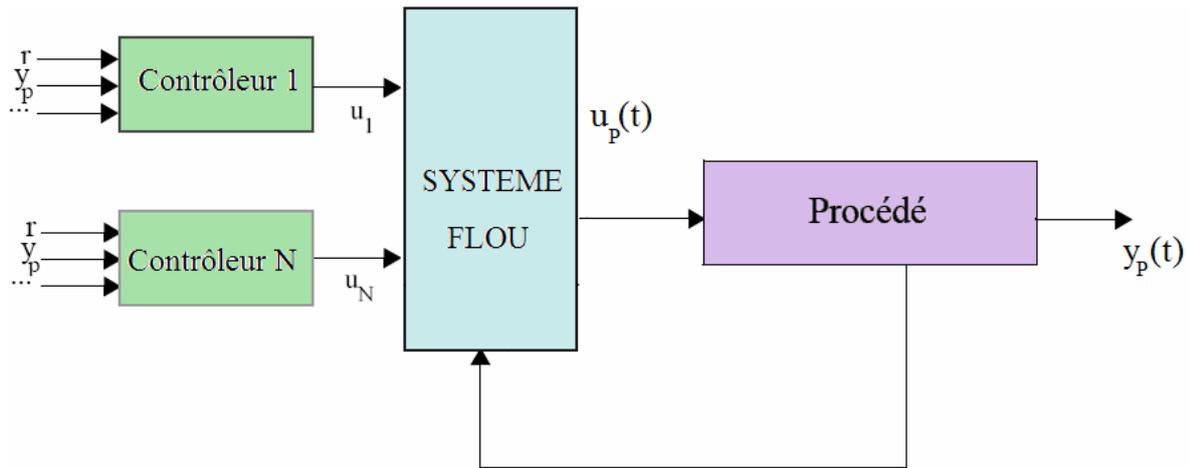


Figure III.9 : Principe de la commutation floue.

Une pondération des lois de commandes locales est fournie par un système flou de type *Takagi-Sugeno* sur des contrôleurs linéaires exposés au prochain chapitre. La variable permettant la commutation (l'entrée du système flou) se trouve être la sortie du procédé. Les contrôleurs sont calculés autour de points de fonctionnement. Autour de chaque point de fonctionnement est défini un « voisinage ». Des fonctions d'appartenance triangulaire sont associées aux zones de fonctionnement définies. Un système flou, vient alors pondérer toutes les lois de commandes locales.

### III.7 Modèle Flou TS

Le modèle flou de Takagi-Sugeno (TS) d'un système dynamique décrit par un ensemble de règles floues *Si-Alors*. Sa caractéristique principale est qu'il représente localement les relations entrées/sorties d'un système en exprimant chaque conclusion par un système linéaire. La  $i^{\text{ème}}$  règle du modèle flou s'écrit:

$$\text{Si } z_1(t) \text{ est } F_i^1 \text{ et...et } z_p(t) \text{ est } F_i^p \text{ Alors : } \begin{cases} \dot{x}(t) = A_i x(t) + B_i u(t) \\ y(t) = C_i x(t) \end{cases} \quad (\text{III.10})$$

$F_i^j$  est l'ensemble flou et  $r$  est le nombre de règles *Si-Alors*.

$x(t) \in R^n$  représente le vecteur d'état,

$u(t) \in R^m$  est le vecteur de commande,

$y(t) \in R^p$  est le vecteur de sortie du système.

$A_i \in R^{n \times n}$  est la matrice d'état,

$B_i \in R^{n \times m}$  est la matrice d'entrée du système,

$C_i \in R^{q \times n}$  est la matrice de sortie,

et  $z_1(t) \dots z_p(t)$  sont les prémisses fonction d'état.

Pour une paire  $(y(t), u(t))$  l'inférence du système flou est donnée par :

$$\left\{ \begin{array}{l} \dot{x}(t) = \frac{\sum_{i=1}^r w_i(z(t))\{A_i x(t) + B_i u(t)\}}{\sum_{i=1}^r w_i(z(t))} \\ y(t) = \frac{\sum_{i=1}^r w_i(z(t))\{C_i x(t)\}}{\sum_{i=1}^r w_i(z(t))} \end{array} \right. \quad (\text{III.11})$$

Où  $z(t)=[z_1(t) z_2(t) \dots z_p(t)]$ ,  $w_i(t) = \prod_{j=1}^p F_i^j(z_j(t))$ ,  $i = 1, 2, \dots, r$

$F_i^j(z_j(t))$  est la valeur de la fonction d'appartenance de  $z_j(t)$  dans l'ensemble flou  $F_i^j$ , et

$\forall t \geq 0$  on a :

$$\left\{ \begin{array}{l} \sum_{i=1}^r w_i(z(t)) > 0 \\ w_i(z(t)) \geq 0 \end{array} \right.$$

En posant :

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

Le modèle flou TS (III.10) s'écrit :

$$\left\{ \begin{array}{l} \dot{x}(t) = \sum_{i=1}^r h_i(z(t))\{A_i x(t) + B_i u(t)\} \\ y(t) = \sum_{i=1}^r h_i(z(t))\{C_i x(t)\} \end{array} \right. \quad (\text{III.12})$$

Les  $h_i(z(t)) \geq 0$  possèdent la propriété de somme convexe :

$$\forall t \geq 0 \quad \sum_{i=1}^r h_i(z(t)) = 1 \quad (\text{III.13})$$

### III.8 Construction d'un modèle flou TS

Pour obtenir un modèle flou TS, deux méthodes distinctes peuvent être employées :

- La première dite par identification, permet à partir des données sur les entrées et les sorties, d'identifier les paramètres du modèle local correspondant aux différents points de fonctionnement ;
- La seconde méthode consiste à linéariser le modèle autour d'un ensemble de points de fonctionnement choisis judicieusement. Dans ce cas, il s'agit de modèles locaux affines pour lesquels le modèle flou est obtenu par interpolation des modèles locaux avec des fonctions d'activation conçu de manière judicieuse en fonctions des spécifications souhaitées (triangulaire, gaussienne,...etc.)

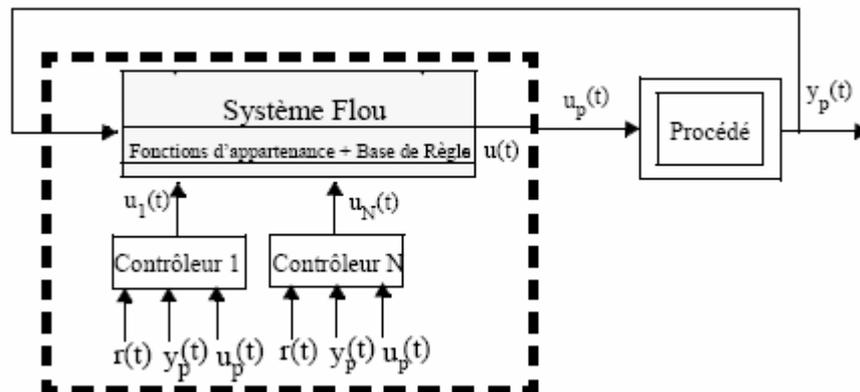


Figure III.10 : Système flou TS (approche directe).

### III.7 Conclusion

En plus de l'approche directe ou indirecte qui détermine la présence ou non du superviseur, deux nouvelles classes peuvent être définies au niveau de la commutation : sélection d'un seul contrôleur ou bien sélection de plusieurs contrôleurs dans la synthèse du signal de commande final.

Après avoir procédé à l'exposé des principales méthodes relatives à la commande multi contrôleurs, l'objectif du prochain chapitre est de donner un rappel théorique succinct relatif aux commandes utilisées pour la synthèse des contrôleurs locaux.

## Chapitre IV

### *Rappel théorique sur les commandes utilisées*

#### IV.1 Introduction

Dans ce chapitre, nous allons aborder les développements théoriques des différentes techniques de commande qui seront utilisées ultérieurement pour synthétiser les régulateurs de la commande multi contrôleurs à appliquer à notre système.

Nous commencerons par la commande  $H^\infty$ , passerons à la commande  $RST$ , puis à la commande par retour d'état et terminerons par la logique floue qui sera utilisée par la suite pour synthétiser un commutateur floue. Celui-ci combinera les effets des régulateurs pour obtenir un signal de commande à appliquer au système.

#### IV.2 La commande $H^\infty$ [21]

La synthèse  $H^\infty$  prend une place de plus en plus importante parmi les méthodes de calcul de correcteur. Sa principale caractéristique est de permettre de modéliser différents transferts d'un système asservi, tout en garantissant la stabilité de la boucle fermée. Elle permet également la prise en compte de certains objectifs de robustesse, tels que la garantie de marges de stabilité ou la robustesse aux dynamiques hautes fréquences mal connues ou non modélisées.

##### IV.2.1 La norme $H^\infty$

Tout système linéaire et invariant peut être modélisé par une fonction de transfert ou transmittance  $G(s)$ . Si le système a  $q$  entrées et  $r$  sorties,  $G(s)$  est une matrice  $r \times q$ . Dans le cas où le système est stable, on définit sa norme  $H^\infty$  par :

$$\|G(s)\|_\infty = \sup_{\omega \in R} \sqrt{\lambda(G(j\omega)G(-j\omega)^T)} \quad (\text{IV.1})$$

Où la notation  $\bar{\lambda}(\ )$  désigne la plus grande des valeurs propres.

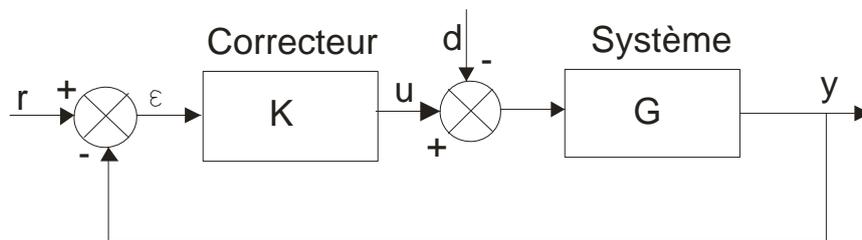
Pour un système SISO, cette expression devient :

$$\|g(s)\|_{\infty} = \sup_{\omega \in \mathbb{R}} |g(j\omega)| \quad (\text{IV.2})$$

La norme  $H^{\infty}$  apparaît donc comme le maximum du module de la fonction de transfert pour les systèmes SISO, et le maximum du module de réponse fréquentielle de la plus grande valeur singulière de  $G(s)$  dans le cas des systèmes MIMO.

### IV.2.2 Analyse des principaux transferts d'un asservissement

Considérons le schéma bloc de la figure (IV.1), où  $G(s)$  est un modèle sous forme matrice de fonctions de transfert du système à asservir, et  $K(s)$  est la matrice des correcteurs ;  $r$  et  $d$  sont respectivement le vecteur des références et le vecteur des perturbations reçues en entrée du système,  $u$  est le vecteur des commandes élaborées par le correcteur,  $y$  est le vecteur des sorties et  $\varepsilon$  est l'erreur d'asservissement.



**Figure IV.1 :** structure d'un asservissement classique

On peut à partir de ce schéma, exprimer la transformée de Laplace de l'erreur en fonction des transformées de Laplace des signaux appliqués :

$$\varepsilon(s) = r(s) - G(s)(-d(s) + K(s)\varepsilon(s))$$

D'où :

$$\varepsilon(s) = \frac{1}{1 + G(s)K(s)} r(s) + \frac{G(s)}{1 + G(s)K(s)} d(s) = T_{er}(s)r(s) + T_{ed}(s)d(s) \quad (\text{IV.3})$$

Et pour la commande, on en déduit :

$$u(s) = \frac{K(s)}{1 + G(s)K(s)}r(s) + \frac{G(s)K(s)}{1 + G(s)K(s)}d(s) = T_{ur}(s)r(s) + T_{ud}(s)d(s) \quad (IV.4)$$

On met ainsi en évidence 4 transferts différents. L'étude de leur réponse fréquentielle renseigne sur les propriétés de l'asservissement : par exemple l'asservissement est sans erreur statique si les transferts  $T_{er}(s)$  et  $T_{ed}(s)$  sont nuls en  $s=0$ .

Par ailleurs, on peut déduire très simplement l'allure de ces réponses fréquentielles (Figure IV.2), en faisant des hypothèses sur le gain de la boucle ouverte :

- dans les zones de grand gain, qui correspondent en général aux basses fréquences (bien en deçà de la bande passante de l'asservissement),  $|G(j\omega)K(j\omega)| \gg 1$  et donc  $1 + G(j\omega)K(j\omega) \approx G(j\omega)K(j\omega)$ . Les approximations reportées sur la figure montrent que le correcteur agit sur les transferts de  $r$  vers  $\varepsilon$  et de  $d$  vers  $\varepsilon$ , tandis qu'il est sans effet sur les transferts de  $r$  vers  $u$  et de  $d$  vers  $u$ ;
- dans les zones de faible gain, qui correspondent en général aux hautes fréquences (bien au-delà de la bande passante de l'asservissement),  $|G(j\omega)K(j\omega)| \ll 1$  et donc  $1 + G(j\omega)K(j\omega) \approx 1$ . Le correcteur agit sur les transferts de  $r$  vers  $u$  et de  $d$  vers  $u$ , tandis qu'il est sans effet sur les transferts de  $r$  vers  $\varepsilon$  et de  $d$  vers  $\varepsilon$ .

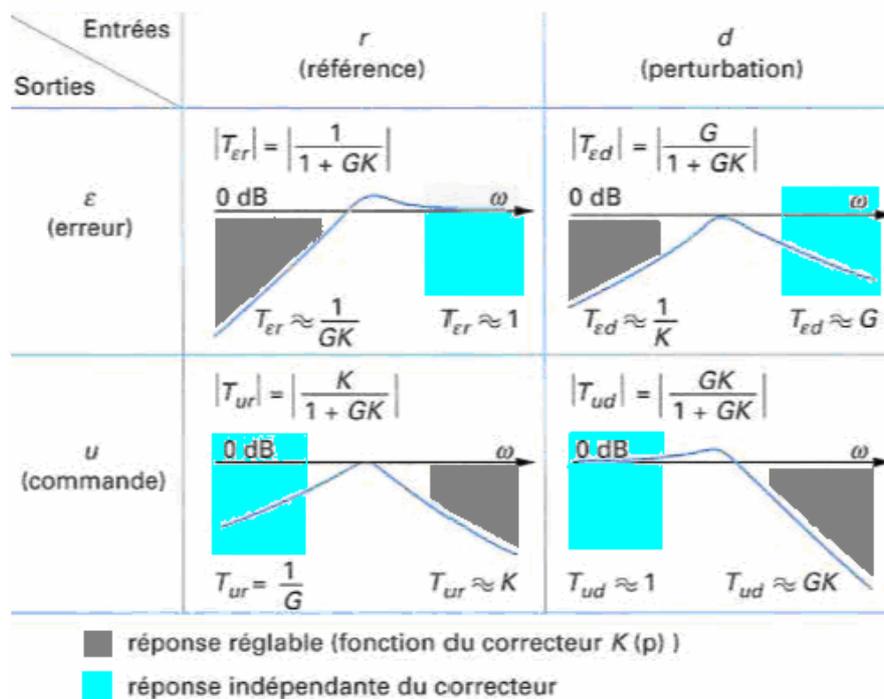


Figure IV.2 : comportement asymptotique des transferts en boucle fermée

### IV.2.3 Le problème $H^\infty$ standard

Le cadre général utilisé pour la synthèse d'un correcteur en utilisant la norme  $H^\infty$  est celui de la figure (IV.3) : le vecteur  $v$  représente toutes les entrées extérieures appliquées à l'asservissement, tel que signaux de référence, perturbations, bruits. Le vecteur  $u$  représente les commandes. Le vecteur  $e$  représente tous les signaux dont on cherche à surveiller l'évolution, et qu'on souhaite maintenir à 0 malgré les entrées  $v$ . Enfin le vecteur  $z$  représente les mesures disponibles pour élaborer la commande.

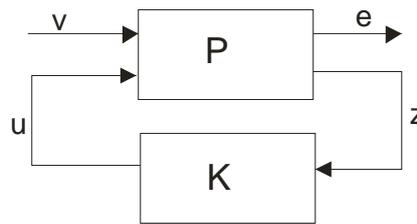


Figure IV.3 : problème  $H^\infty$  standard

La matrice de transfert  $P(s)$  qui apparaît sur ce schéma représente les interconnexions entre les signaux  $v$  et  $u$  d'une part,  $e$  et  $z$  d'autre part, tandis que  $K(s)$  désigne le correcteur que l'on cherche à calculer.

Appelons  $T_{ev}(s)$  la matrice de transfert entre les signaux  $v$  et  $e$  sur la figure. La synthèse du correcteur peut être menée en cherchant :

- à assurer la stabilité du système bouclé ;
- à limiter le gain entre  $v$  et  $e$  à une certaine valeur  $\gamma$ , donc à assurer à  $T_{ev}(s)$  une norme  $H^\infty$  inférieure à  $\gamma$ .

Ce problème est appelé le problème  $H^\infty$  standard. Les correcteurs correspondant à la valeur de  $\gamma$  la plus petite possible seront dits « optimaux ».

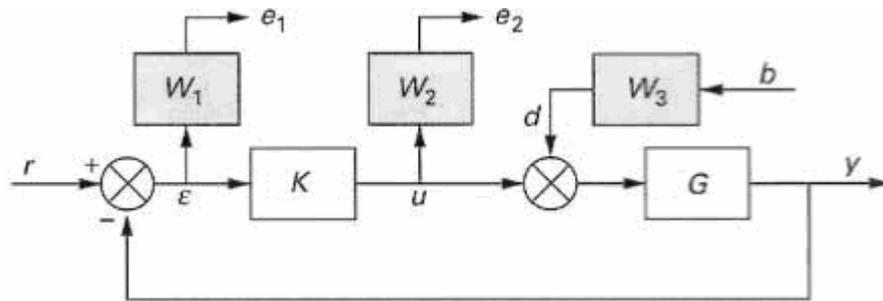
La méthode de résolution la plus utilisée pour déterminer un correcteur solution du problème est l'algorithme de Glover-Doyle. Il peut être brièvement décrit comme suit :

- à partir de la résolution de deux équations matricielles non linéaires (appelées équations de Riccati), on peut savoir si la norme  $H^\infty$  de  $T_{ev}(s)$  peut être inférieure à une valeur  $\gamma$  donnée ; la valeur minimale de  $\gamma$  est approchée par itérations successives ;
- pour toute valeur  $\gamma$  admissible, il existe un correcteur  $K(s)$  solution du problème, dont l'ordre est égal à celui de  $P(s)$ .

**IV.2.4 Mise en œuvre par le choix de pondérations fréquentielles**

L'un des problèmes les plus fréquemment considérés en synthèse  $H^\infty$  est illustré par le schéma bloc de la figure (VI.4) : en appliquant des filtres  $W_1(s)$ ,  $W_2(s)$ ,  $W_3(s)$  aux sorties  $\varepsilon$  et  $u$  et à la perturbation  $d$  respectivement, on introduit trois nouveaux signaux  $e_1$ ,  $e_2$  et  $b$  qui vérifient :

$$\begin{aligned} e_1(s) &= W_1(s)\varepsilon(s) \\ e_2(s) &= W_2(s)u(s) \\ d(s) &= W_3(s)b(s) \end{aligned} \tag{IV.5}$$



**Figure IV.4 :** exemple de mise en place d'une synthèse  $H^\infty$

On cherche alors un correcteur  $K(s)$  stabilisant la boucle fermée, et assurant à la matrice de transfert entre les entrées  $r, b$  et les sorties  $e_1, e_2$  une norme  $H^\infty$  inférieure à 1 :

$$\left\| \begin{pmatrix} T_{e_1 r}(s) & T_{e_1 b}(s) \\ T_{e_2 r}(s) & T_{e_2 b}(s) \end{pmatrix} \right\|_\infty = \left\| \begin{pmatrix} W_1(s)T_{\varepsilon r}(s) & W_1(s)T_{\varepsilon d}(s)W_3(s) \\ W_2(s)T_{ur}(s) & W_2(s)T_{ud}(s)W_3(s) \end{pmatrix} \right\|_\infty < 1 \tag{IV.6}$$

Une des propriétés de la norme  $H^\infty$  est que si l'inégalité (IV.7) est vérifiée, alors chacun des différents transferts composant la matrice possède la même propriété. On aura ainsi :

$$\|W_1(s)T_{\varepsilon r}(s)\|_\infty < 1 \Leftrightarrow |W_1(j\omega)T_{\varepsilon r}(j\omega)| < 1 \Leftrightarrow |T_{\varepsilon r}(j\omega)| < \frac{1}{|W_1(j\omega)|} \tag{IV.7}$$

Et de même :

$$\begin{aligned} |T_{ur}(j\omega)| &< \frac{1}{|W_2(j\omega)|} \\ |T_{\varepsilon d}(j\omega)| &< \frac{1}{|W_1(j\omega)| \cdot |W_3(j\omega)|} \\ |T_{ud}(j\omega)| &< \frac{1}{|W_2(j\omega)| \cdot |W_3(j\omega)|} \end{aligned} \tag{IV.8}$$

Ainsi, par le choix des filtres  $W_1(s)$ ,  $W_2(s)$  et  $W_3(s)$ , l'utilisateur peut agir sur les réponses fréquentielles des quatre transferts  $T_{er}(s)$ ,  $T_{ur}(s)$ ,  $T_{ed}(s)$  et  $T_{ud}(s)$ . Les principes généraux guidant ses choix, sont les suivants :

- le gain  $|T_{er}(j\omega)|$  doit être faible en basse fréquence, pour assurer les objectifs de précision. Sa fréquence de passage à 0 dB correspond, à peu près, à la bande passante de l'asservissement. Son maximum, c'est-à-dire sa norme  $H_\infty$  est l'inverse de la marge de module. Enfin, aucune contrainte ne peut être imposée à  $T_{er}(s)$  en haute fréquence, où son gain tend vers 1 indépendamment du correcteur ;
- le gain  $|T_{ur}(j\omega)|$  ne peut être contraint en basse fréquence. Il doit être faible en haute fréquence, au-delà de la bande passante choisie pour l'asservissement, pour éviter de solliciter inutilement la commande, ou d'exciter des dynamiques mal modélisées ;
- le gain  $|T_{ed}(j\omega)|$  doit être faible en basses et moyennes fréquences, pour assurer une réjection correcte de la perturbation. Il ne peut être contraint en haute fréquence ;
- le gain  $|T_{ud}(j\omega)|$  ne peut être contraint qu'en moyennes et hautes fréquences. Sa fréquence de passage à 0 dB correspond à peu près à la bande passante de l'asservissement. Il doit être faible en haute fréquence, pour éviter d'exciter inutilement la commande.

### IV.3 La commande RST

Cette partie est consacrée à la description de la synthèse d'un régulateur à deux degrés de liberté polynomial robuste type *RST*. Il est souvent utilisé en échantillonné. Les polynômes R S et T sont notés dans certains ouvrages *L*, *K* et *H* respectivement.

### IV.3.1 Approche décentralisé

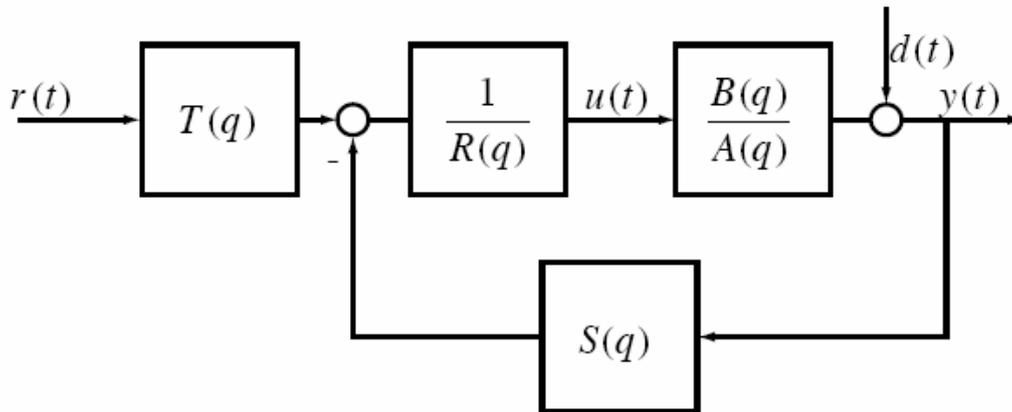


Figure IV.5 : schéma bloc de la commande RST

#### IV.3.1.1 modèle du système

Le processus est un système (SISO) ayant pour entrée de commande la grandeur  $u(t)$ , et pour sortie  $y(t)$ . Il est modélisée par une fonction de transfert  $H(s) = \frac{B(s)}{A(s)}$  strictement propre entre la commande et la sortie. Le caractère strictement propre ( $\deg A > \deg B$ ) assure un retard d'au moins une période d'échantillonnage sur la sortie ce qui sera nécessaire pour permettre la synthèse du régulateur. Tout système physique possède un retard strictement positif. Les polynômes A et B sont choisis entiers entre eux. A la sortie du système s'ajoute une perturbation  $d(t)$  non mesurable directement. La sortie complète est donc :

$$y(s) = \frac{B(s)}{A(s)}u(s) + d(s) \quad (\text{IV.9})$$

#### IV.3.1.2 Structure du régulateur

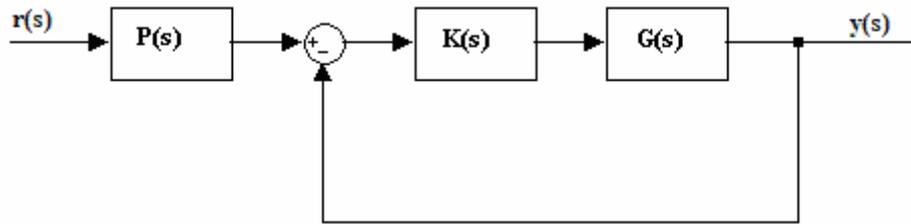
Il a la forme suivante (voir la figure IV.4) :

$$R \cdot u(S) = -S \cdot y(S) + T \cdot r(S) \quad (\text{IV.10})$$

Les degrés de liberté se réfèrent à la prise en compte différenciée de la sortie et de la consigne pour l'élaboration du signal de commande. Dans un régulateur à un degré de liberté, la commande se base sur le signal d'erreur  $e(t) = r(t) - y(t)$ , ce qui force à choisir un compromis entre la stabilité et le rejet des perturbations d'une part, et l'asservissement à une grandeur de consigne, d'autre part. [10]

**Remarque :**

La structure RST est équivalente à une boucle de régulation classique + un pré compensateur.



**Figure IV.6 :** schéma équivalent de la commande RST

Avec :

$$P(s) = \frac{T(s)}{S(s)} \quad \text{et} \quad K(s) = \frac{S(s)}{R(s)} \quad (\text{IV.11})$$

Pour que le régulateur soit causal, les conditions suivantes doivent être remplies :

$$\begin{aligned} \deg R &\geq \deg S \\ \deg R &\geq \deg T \end{aligned} \quad (\text{IV.12})$$

En boucle fermée, la fonction de transfert  $H_c(s)$  entre la consigne et la sortie est :

$$H_c(s) = \frac{B(s) \cdot T(s)}{A(s) \cdot R(s) + B(s) \cdot S(s)} \quad (\text{IV.13})$$

et la fonction de transfert  $H_p(s)$  entre la perturbation  $d(s)$  et la sortie  $y(s)$  :

$$H_p(s) = \frac{A(s) \cdot R(s)}{A(s) \cdot R(s) + B(s) \cdot S(s)} \quad (\text{IV.14})$$

La sortie vaut alors :

$$y(s) = H_c(s) \cdot r(s) + H_p(s) \cdot d(s) \quad (\text{IV.15})$$

Une façon répandue [11] de fixer les polynômes d'un régulateur RST consiste à imposer l'égalité  $BT/(AR + BS) = Bm/Am$ , où  $Bm/Am$  est la fonction de transfert désirée en asservissement.

On peut dès lors introduire une simplification entre les zéros des différents polynômes. En imposant un facteur commun  $A_0$  entre  $(AR + BS)$  et  $T$  (couramment appelé *Polynôme observateur* par analogie avec la commande d'état), on fait disparaître une partie de la dynamique interne de la fonction de transfert en asservissement, qui reste présente en régulation. Dans le cadre de ce travail, aucune de ces simplifications n'est envisagée. Elles ne

sont en effet valables que lorsque le système est exactement identique au modèle, hypothèse difficilement justifiable en commande robuste.

D'autre part, nous nous intéressons explicitement au comportement en régulation et nous voulons prendre en compte le signal de commande. [10]

Le dénominateur  $C(s)$  de  $Hc(s)$  et de  $Hp(s)$ , identique, est le *polynôme caractéristique*, dont les zéros sont les *pôles en boucle fermée* :

$$C(s) = A(s)R(s) + B(s)S(s) \quad (\text{IV.16})$$

Le polynôme caractéristique détermine la stabilité du système en boucle fermée et contribue largement à ses propriétés dynamiques. Celles-ci dépendent également des zéros des fonctions de transfert entre différents types de perturbations et la commande ou la sortie, mais on ne peut pas les spécifier indépendamment du polynôme caractéristique

#### IV.3.1.3 placement de pôle

Une façon de fixer la valeur des polynômes  $R(s)$  et  $S(s)$  est d'imposer un certain polynôme caractéristique  $C(s)$ . On peut aussi prédéfinir un modèle de référence dont on y spécifie nos performances et robustesse, ce modèle de référence imposera ces pôles au système bouclé :

$$y_m(s) = \frac{B_m(s)}{A_m(s)} r(s) \quad (\text{IV.17})$$

Pour que la sortie  $y(s)$  suit le modèle de référence il faut que :

$$\frac{B_m(s)}{A_m(s)} = \frac{B(s)T(s)}{C(s)} \quad (\text{IV.18})$$

Comme  $B(s)$  et  $A(s)$  sont figés (caractéristiques du système) on peut calculer les polynômes  $R$ ,  $S$  et  $T$ . les équations peuvent être sous cette forme :

$$\begin{cases} T(s) = \frac{B_m(s)}{B(s)} \\ A(s)R(s) + B(s)S(s) = A_m(s) \end{cases} \quad (\text{IV.19})$$

Mais comme  $R(s)$ ,  $S(s)$  et  $T(s)$  sont des polynômes et non pas des fonctions de transfert, on obtient ainsi l'équation DIOPHANTINE (du nom du mathématicien Diophante, IIIe siècle) suivante :

$$\begin{cases} T(s) = B_m(s) \\ A(s)R(s) + B(s)S(s) = C(s) = A_m(s)B(s) \end{cases} \quad (\text{IV.20})$$

L'équation diophantienne (IV.20) est linéaire par rapport aux coefficients de  $R(s)$  et  $S(s)$ . En imposant l'égalité des coefficients des différentes puissances de  $s$ , on obtient l'équation matricielle :  $Sy = b$ .

Le vecteur  $x$  contient les coefficients des polynômes  $R(s)$ , et  $S(s)$ . La matrice  $S_y$  s'appelle la matrice de Sylvester. Il faut choisir le degré des polynômes et du modèle de référence de manière à avoir  $S_y$  de rang plein afin d'obtenir une solution unique à l'équation (IV.20)

#### IV.3.1.4 filtrage de la consigne

En plus de la partie contre-réaction, le régulateur à deux degrés de liberté comporte un polynôme  $T(s)$  qui filtre spécifiquement la consigne; nous l'appellerons *pré filtre*. Par rapport aux régulateurs classiques où seule la différence entre la consigne et la sortie est prise en compte pour l'élaboration de la grandeur de commande, cela offre les avantages suivants :

- Le gain statique de la fonction de transfert en boucle fermée  $BT/(AR + BS)$  dépend directement de  $T$ . En fixant  $T(0)/(A(0)R(0) + B(0)S(0)) = 1$ , on garantit l'absence de statisme vis-à-vis d'une consigne constante ;
- Lorsque tous les pôles sont réels, des bornes sur le nombre d'extrema peuvent être déterminées en fonction de la position des zéros, ce qui permet dans certains cas de garantir l'absence de dépassement [12] ;
- Un comportement vicieux, où la réponse transitoire démarre dans le sens opposé à la consigne, peut être atténué par l'ajout de zéros supplémentaires au prix d'un retard de la sortie ou d'une anticipation de la consigne. [10] ;

De façon plus générale, le filtre de consigne permet d'ajuster le dépassement, de contrer le dépassement négatif, de gérer le temps de montée, de définir le temps d'établissement, et d'amoindrir les effets transitoires sur la commande.

Pour notre régulateur RST, le choix du filtre de consigne sera fait lors de la sélection du modèle de référence.

### IV.3.2 Approche centralisé

Cette approche polynomiale convient à la synthèse mono variable (système découplé), mais si on désire synthétiser en multi variables, on doit procéder différemment.

Etant donnée  $G(s)$  une matrice de fonctions de transfert, on peut définir 8 MFT ( $A(s)$ ,  $B(s)$ ,  $Ab(s)$ ,  $Bb(s)$ ,  $Fad(s)$ ,  $Fbd(s)$ ,  $Fag(s)$ ,  $Fbg(s)$ ) propres et stables tel que :

$G(s)=B(s).A^{-1}(s)$  on l'appelle : factorisation co-première à droite

$G(s)=Ab^{-1}(s) Bb(s)$  qu'on appelle : factorisation co-première à gauche.

$$Fad(s)*A(s) + Fbd(s)*B(s)=I$$

On ne s'intéressera qu'aux 6 premières matrices qui nous permettrons de calculer  $H(s)$ ,  $L(s)$  et  $K(s)$  du régulateur, et on les trouve de la manière suivante:

Soit  $(A_1, B_1, C_1, D_1)$  les matrices de la représentation d'état :

On détermine 2 matrices quelconques  $F$  et  $h$  tel que :  $A_1+B_1k$  et  $A_1+hC_1$  soient stables (on peut les déterminer par placement de pôles).

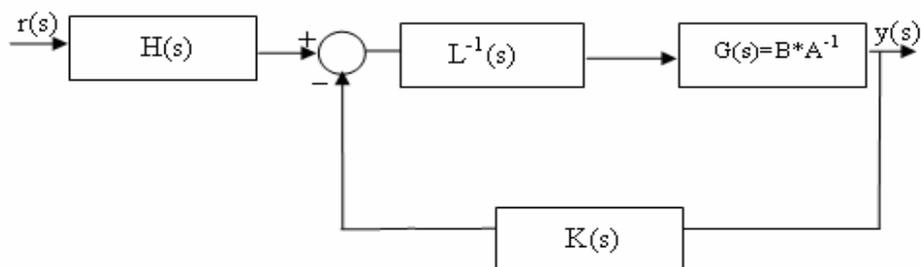
Ensuite les représentations d'état des différentes matrices sont données par le tableau suivant :

**Tableau IV.1 :** Représentation d'état des différentes matrices servant au calcul du RST centralisé

$B(s) = \begin{bmatrix} A_1 - B_1F & B_1 \\ C_1 - D_1F & D_1 \end{bmatrix}$	$A(s) = \begin{bmatrix} A_1 - B_1F & B_1 \\ -F & I \end{bmatrix}$	$F_{db}(s) = \begin{bmatrix} A_1 - hC_1 & h \\ F & D_1 \end{bmatrix}$
$F_{ad}(s) = \begin{bmatrix} A_1 - hC_1 & B_1 \\ F & I \end{bmatrix}$	$B_d(s) = \begin{bmatrix} A_1 - hC_1 & B_1 - hD_1 \\ C_1 & D_1 \end{bmatrix}$	$A_d(s) = \begin{bmatrix} A_1 - hC_1 & -h \\ C_1 & I \end{bmatrix}$

Par la suite, on procède de la manière suivante:

Soit la matrice de transfert  $G(s)$  factorisé à droite :



**Figure IV.7 :** Commande RST centralisée

On choisit un modèle de référence  $T_y^d(s) = A_m^{-1}(s) \cdot B_m(s)$ , qu'on a factorisé à gauche :

On a  $T_y = B[LA + KB]^{-1}H$  (la fonction de transfert en boucle fermée)

La poursuite du modèle de référence  $T_y^d$  est obtenue en prenant le choix suivant :

$$\begin{cases} H(s) = B_m(s) \\ L(s) = A_m(s)B(s)F_{ad}(s) \\ K(s) = A_m(s)B(s)F_{bd}(s) \end{cases} \quad (\text{IV.21})$$

#### IV.4 Commande par retour d'état

La commande par retour d'état est une méthode employée en asservissement pour placer les pôles en boucle fermée dans le plan  $s$ . L'intérêt de cette technique est que les pôles, correspondant à la valeur propre du système, influencent la dynamique du système bouclé, et commandent ainsi la réponse du système.

Etant donnée la représentation d'état d'un système  $(\Sigma)$  à  $m$  entrées et  $p$  sorties :

$$\begin{cases} \dot{X} = AX + BU \\ Y = CX \end{cases} \quad (\text{IV.22})$$

Avec :

$X \in R^n$  vecteur d'état de dimension  $n$

$U \in R^m$  vecteur de commande de dimension  $m$

$Y \in R^p$  vecteur de sortie de dimension  $p$

Alors :

$A$  est une matrice  $n \times n$

$B$  est une matrice  $n \times m$

$C$  est une matrice  $p \times n$

Alors les pôles du système sont les racines de l'équation :

$$\det(sI - A) = 0 \quad (\text{IV.23})$$

La commande par retour d'état s'effectue directement sur le vecteur entrée  $\underline{u}$ . On a alors une entrée proportionnelle (au sens matriciel) à l'état :

$$\underline{u} = -K\underline{x} + \underline{v} \quad (\text{IV.24})$$

Ceci remplacé dans les équations (IV.22) :

$$\begin{cases} \dot{X} = (A - BK)X + BV \\ Y = CX \end{cases}$$

Où  $V$  est le vecteur consigne.

Les racines du système bouclé sont données par l'équation  $\det(sI - (A - BK))$ . Ce terme doit être égalé à celui du polynôme caractéristique de l'asservissement désiré. On obtient alors les coefficients de la matrice de retour  $K$  qui va imposer les valeurs propres en boucle fermée.

Ceci est possible si le système est complètement commandable, c'est à dire :

$$\text{rang}(C_0) = \text{rang} \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix} = n$$

Dans le cas où la matrice de commandabilité  $C_0$  n'est pas de rang complet, on ne peut placer que les modes commandables, les modes non commandables étant fixes.

Cependant, une erreur statique peut exister. Pour y remédier, une action intégrale est ajoutée pour éliminer cette erreur. La commande devient alors :

$$U = -K.X - G.\int (C.X - V)dt \quad (\text{IV.25})$$

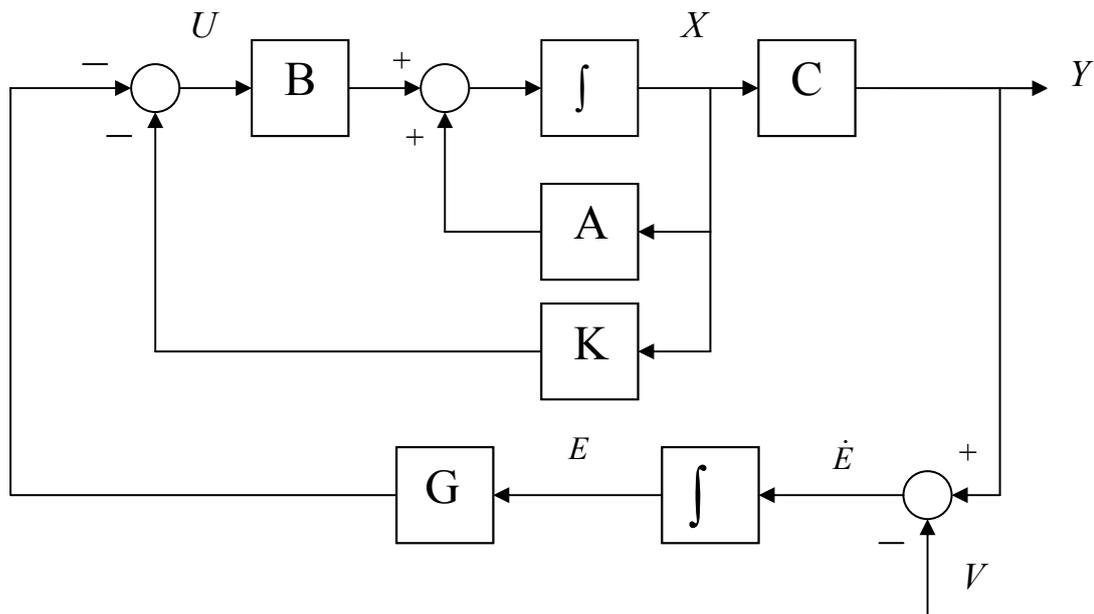
$K$  et  $G$  sont les matrices gain à déterminer.

On pose :

$$\varepsilon = \int (C.X - V)dt \Rightarrow \dot{\varepsilon} = (C.X - V) \quad (\text{IV.26})$$

Ainsi on obtient la représentation d'état augmentée suivante :

$$\begin{cases} \begin{pmatrix} \dot{X} \\ \dot{\varepsilon} \end{pmatrix} = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix} \begin{pmatrix} X \\ \varepsilon \end{pmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} U + \begin{bmatrix} 0 \\ -I \end{bmatrix} V \\ Y = [C \quad 0] \begin{pmatrix} X \\ \varepsilon \end{pmatrix} \\ U = -[K \quad G] \begin{pmatrix} X \\ \varepsilon \end{pmatrix} \end{cases} \quad (\text{IV.27})$$



**Figure IV.8 :** schéma bloc de la commande par retour d'état avec action intégrale

On note :

$$A_{aug} = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix}$$

$$B_{aug} = \begin{bmatrix} B \\ 0 \end{bmatrix}$$

Un placement de pôle est effectué pour déterminer les matrices gain  $K$  et  $G$ .

La condition de commandabilité sur la paire  $(A_{aug}, B_{aug})$  est toujours nécessaire.

## IV.5 La logique floue

### IV.5.1 Principe et définitions [17]

La logique floue repose sur la théorie des ensembles flous développée par Zadeh. Face à un formalisme mathématique fort développé, nous préférons aborder la présentation de manière intuitive.

Les notions de température moyenne ou de courant faible sont relativement difficiles à spécifier de manière précise. On peut fixer des seuils et considérer que l'on attribue tel ou tel qualificatif en fonction de la valeur de la variable par rapport à ces seuils. Ceci ne peut exprimer qu'un avis très « tranche » du qualificatif "température moyenne" par exemple. L'aspect "vague" de ce qualificatif n'est pas représenté (figure IV.8).

On peut définir le degré d'appartenance de la variable température à l'ensemble "faible" comme le "degré de vérité" de la proposition "la température est faible".

En logique booléenne, le degré d'appartenance ( $\mu$ ) ne peut prendre que deux valeurs (0 ou 1). La température peut être:

- Faible :  $\mu_{faible}(T) = 1$  ;  $\mu_{moyenne}(T) = 0$  ;  $\mu_{elevée}(T) = 0$
- Moyenne :  $\mu_{faible}(T) = 0$  ;  $\mu_{moyenne}(T) = 1$  ;  $\mu_{elevée}(T) = 0$
- Elevée :  $\mu_{faible}(T) = 0$  ;  $\mu_{moyenne}(T) = 0$  ;  $\mu_{elevée}(T) = 1$

Elle ne peut pas prendre deux qualificatifs à la fois.

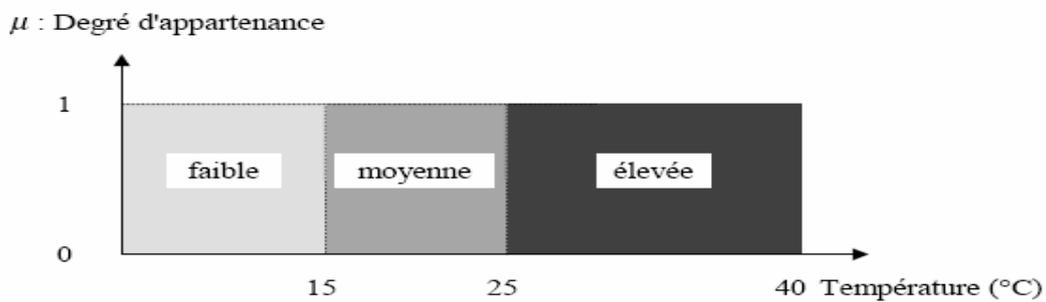


Figure IV.9 : exemple d'ensembles considérés en logique booléenne

En logique floue, le degré d'appartenance devient une fonction qui peut prendre une valeur réelle comprise entre 0 et 1 inclus.

$\mu_{moyenne}(T)$ , par exemple, permet de quantifier le fait que la température puisse être considérée comme moyenne.

Dans ce cas, la température peut être considérée, à la fois, comme faible avec un degré d'appartenance de 0,2 et comme moyenne avec un degré d'appartenance de 0,8 (figure IV.9).

$$\mu_{faible}(T) = 0.2 \quad ; \quad \mu_{moyenne}(T) = 0.8 \quad ; \quad \mu_{elevée}(T) = 0$$

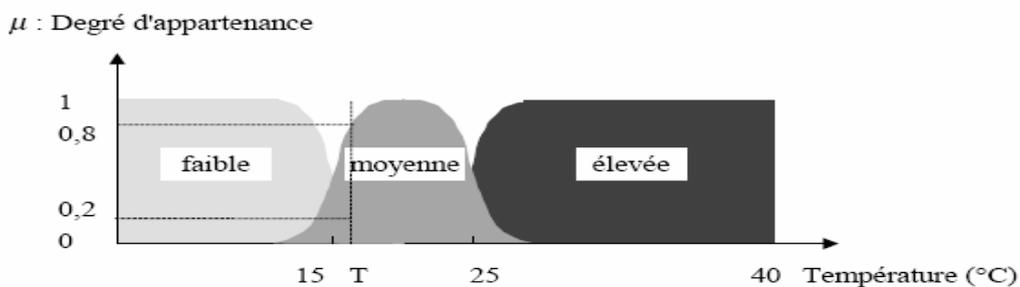


Figure IV.10 : exemple d'ensembles considérés en logique floue

### IV.5.2 Ensemble flou

Soit  $X$  une collection continue ou discrète d'objets dénotés  $\{X\}$ ,  $X$  est appelé univers de discours et  $x$  représente l'élément générique de  $X$ .

Un ensemble flou peut être vu comme une généralisation du concept d'ensemble ordinaire dont la fonction d'appartenance prend seulement deux valeurs  $\{0, 1\}$  (ensemble booléen). Ainsi sur un univers de discours  $X$ , un sous ensemble flou  $A$  est caractérisé par une fonction d'appartenance  $\mu_A$  prenant ses valeurs dans l'intervalle  $[0,1]$ , alors l'ensemble flou  $A$  dans  $X$  est défini par l'ensemble des pairs (élément générique, degrés d'appartenance).

$$A = \{(x, \mu_x(x)) / x \in X\}$$

### IV.5.3 Variables linguistiques

La description d'une certaine situation, d'un phénomène ou d'un procédé contient en général des expressions linguistiques (floues), comme : chaud, froid, grand, petit, ... etc.

Les expressions de ce genre forment les valeurs d'une variable linguistique, qu'on appelle valeurs floues qu'on peut représenter par des fonctions d'appartenances.

### IV.5.4 Fonction d'appartenance

Afin de permettre un traitement numérique des variables linguistiques dans la prise de décisions sur ordinateur, une définition des variables linguistiques à l'aide de fonctions d'appartenance s'impose. Dans ce contexte, on associe à chaque valeur de la variable linguistique une fonction d'appartenance désignée par  $\mu_A(x)$  où  $x$  est la variable linguistique, tandis que  $A$  indique l'ensemble concerné. Une valeur précise de  $\mu_A(x)$ , sera désignée par le degré ou le facteur d'appartenance.

Il est à noter que l'ensemble des éléments de  $x$  pour lesquels  $\mu_A(x) > 0$ , est appelé support de  $A$ .

Le plus souvent, on utilise pour les fonctions d'appartenance les fonctions suivantes :

#### **Fonction triangulaire :**

Elle est définie par trois paramètres  $\{a, b, c\}$ , qui déterminent les coordonnées des trois sommets.

$$\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \quad (\text{IV.28})$$

**Fonction trapézoïdale :**

Elle est définie par quatre paramètres  $\{a, b, c, d\}$

$$\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \quad (\text{IV.29})$$

**Fonction gaussienne :**

Elle est définie par deux paramètres  $\{\sigma, m\}$

$$\mu(x) = \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right) \quad (\text{IV.30})$$

**Fonction sigmoïde :**

Une fonction sigmoïde est définie par deux paramètres  $\{a, c\}$

$$\mu(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (\text{IV.31})$$

**IV.5.5 Raisonnement flou**

Les ensembles flous et la théorie des possibilités sont des éléments importants de la représentation des connaissances imparfaitement définies. Pour raisonner sur de telles connaissances, la logique classique ne suffit pas. On utilise la logique floue lorsque les connaissances sont imprécises, vagues et éventuellement incertaines. Voici quelques définitions utilisées :

Soit  $x$  une variable linguistique et  $A$  une caractéristique :

**Définition 1 (proposition)**

Une proposition floue est définie à partir d'une variable linguistique  $(x, T(x), X)$  par la qualification : "  $x$  est  $A$  ".

**Définition 2 (conjonction)**

La conjonction de deux propositions floues est réalisée par l'opérateur  $ET$  par exemple :

"  $x_1$  est  $A_1$   $ET$   $x_2$  est  $A_2$  "

**Définition 3 (disjonction)**

La disjonction de deux propositions floues est réalisée par l'opérateur *OU* par exemple :

''  $x_1$  est  $A_1$  *OU*  $x_2$  est  $A_2$  ''

**Définition 4 (implication)**

L'implication entre deux propositions floues définit aussi une proposition floue que l'on peut exprimer par :

''  $x_1$  est  $A_1$  *ALORS*  $x_2$  est  $A_2$  ''

Plus généralement, on peut construire des propositions floues par conjonction, disjonction ou implication portant sur des propositions floues élémentaires.

**Définition 5 (règle floue)**

Une règle floue est une proposition floue utilisant une implication entre deux propositions floues quelconques. Par exemple :

'' *SI*  $x_1$  est  $A_1$  *ET*  $x_2$  est  $A_2$  *ALORS*  $x_3$  est  $A_3$  ''

où : ''  $x_1$  est  $A_1$  *ET*  $x_2$  est  $A_2$  '' est la prémisse de la règle

et ''  $x_3$  est  $A_3$  '' est sa conclusion.

En utilisant les règles de composition d'inférence, nous pouvons formaliser une procédure d'inférence, appelé raisonnement flou sur l'ensemble des règles (*Si - Alors*).

**IV.5.6 Commande par la logique floue**

La commande par logique floue est le domaine dans lequel il existe le plus de réalisation effective, en particulier industrielle. Son but est de traiter des problèmes de commande de processus, le plus souvent à partir des connaissances des experts.

**IV.5.7 Contrôleur flou**

La configuration de base d'un contrôleur flou (figure IV. 10) comprend les éléments suivants

**Fuzzification :**

C'est la partie du contrôleur flou chargé de convertir les grandeurs physiques en variables linguistiques.

**Base de règles :**

Elle contient les définitions des termes utilisés dans la commande et l'ensemble des règles caractérisant la cible de la commande et décrivant la conduite de l'expert.

**Inférence :**

L'inférence transforme à l'aide du jeu de règles (en manipulant la base de règles). La partie floue issue de la fuzzification en une nouvelle partie floue qui caractérise la sortie du contrôleur.

**Défuzzification :**

La défuzzification consiste à convertir la partie floue issue de l'inférence en une grandeur physique.

Plusieurs stratégies de défuzzification sont utilisées:

- Méthode du maximum :

La commande est égale à la valeur dont le degré d'appartenance est le plus fort.

- Méthode de la moyenne des maxima :

La commande sera égale à la moyenne des valeurs dont le degré d'appartenance est maximal.

- Méthode du centre de gravité:

C'est la méthode la plus utilisée dans les contrôleurs flous. Dans celle-ci la commande sera égale au centre de gravité de l'ensemble flou de sortie, on obtient donc pour :

-Univers de discours discret

$$C = \frac{\sum_{i=1}^n x_i \mu_A(x_i)}{\sum_{i=1}^n \mu_A(x_i)} \quad (\text{IV.32})$$

-univers de discours continu

$$C = \frac{\int x \cdot \mu_A dx}{\int \mu_A dx} \quad (\text{IV.33})$$

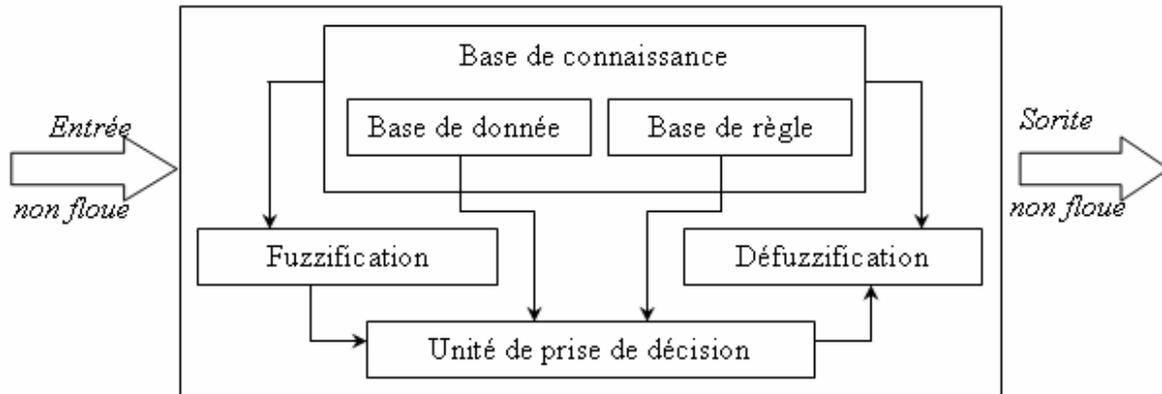


Figure IV.11 : structure de base d'un contrôleur flou

### IV.5.8 Types de régulateurs flous

Il existe plusieurs types de régulateurs flous, qui diffèrent selon le mécanisme d'inférence utilisé. On cite: régulateur de Mamdani, de Sugeno, ... etc.

#### IV.5.8.1 Régulateur de type Mamdani

Mamdani fut le premier à utiliser la logique floue pour la synthèse de commande. Il utilise le minimum comme opérateur de jonction et l'implication pour représenter le graphe flou associé à chaque règle et l'opérateur maximum pour l'agrégation.

Dans la règle  $i$  : *Si  $x_1$  est  $A_1$  et  $x_n$  est  $A_n$  Alors  $y$  est  $B_i$*

où  $B_i$  sont des sous ensembles flous. Les  $B_i$  forment en général une partition de l'univers de sortie.

#### IV.5.8.2 Régulateur de type Sugeno

Dans les régulateurs de ce type, les conclusions des règles ne sont pas symboliques (i.e. représentées par des sous ensembles flous) mais une fonction des entrées, par exemple :  $b_i = f(x_1, \dots, x_n)$  où  $f(\cdot)$  est généralement une fonction polynomiale.

Et la sortie du régulateur est donnée par :

$$y = \frac{\sum_{i=1}^n \alpha_i(x) * b^i}{\sum_{j=1}^n \alpha_j(x)} \quad (\text{IV.34})$$

où les  $\alpha_i$  sont les valeurs de vérité de chaque règle pour  $i=1$  à  $n$ .

Notons que la sortie donnée par le régulateur est en effet la variation du signal de commande.

## **IV.6 Conclusion**

Dans ce chapitre nous avons abordé les développements théoriques des commandes :  $H^\infty$ ,  $RST$ , et la commande par retour avec action intégrale.

Ces techniques seront utilisées par la suite pour synthétiser un régulateur multi contrôleurs afin d'asservir les angles d'élévation et d'azimut du TRMS.

Enfin, on a introduit les notions de la logique floue qui sera utilisée pour développer un commutateur flou dont le rôle est de combiner les effets des différents contrôleurs calculés indépendamment. En effet, ce commutateur permet d'éliminer les sauts dans le signal de commande lors de la commutation d'un contrôleur à un autre.

## Chapitre V

### *Synthèse des commandes*

#### V.1 Introduction

L'objectif de la commande multi contrôleurs est d'asservir (régulation, suivi de trajectoire,...) la sortie du procédé dans tout l'espace de variation des paramètres considérés à l'aide des signaux de commande élaborés par les différents contrôleurs locaux.

Afin de valider nos approches, nous allons synthétiser les commandes dont le développement théorique a été présenté au chapitre précédent pour contrôler l'angle d'élévation et d'azimut du TRMS. La commutation entre les différents contrôleurs se fera via un système à événement discret en premier lieu, ensuite elle sera remplacée par un système flou.

Mais le calcul des régulateurs locaux passe d'abord par le calcul des modèles linéarisés autour de points de fonctionnement choisis.

#### V.2 linéarisation tangente

##### V.2.1 principe

Notre système étant affine en la commande présente une dynamique sous la forme :

$$\begin{cases} \dot{x} = f(x) + g(x).u \\ y = h(x) \end{cases} \quad (\text{V.1})$$

On peut écrire ces équations de façon générale :

$$\begin{cases} \dot{x} = f(x) + g(x, u) \\ y = h(x) \end{cases} \quad (\text{V.2})$$

Selon l'approche analytique, la linéarisation tangente se base sur un développement en série de Taylor des équations du modèle disponible et sur une approximation de ces équations à l'ordre 1. Pour quantifier le modèle linéaire obtenu, ce développement se fait autour d'un point de référence qui doit être un point d'équilibre du système afin de satisfaire l'hypothèse de stationnarité temporelle. Le modèle linéaire obtenu correspondra à une approximation de la dynamique du système au voisinage de ce point d'équilibre.

### V.2.2 points d'équilibre

Les points d'équilibre d'un système (ou ensemble d'équilibre) sont les points où l'équation de la dynamique de ce système s'annule (i.e. les points de  $R^n \times R^m$  où la fonction non-linéaire  $f(x)+g(x,u)$  s'annule).

Cet ensemble est défini par :

$$\varepsilon = \left\{ (x_0, u_0) \in R^n \times R^m / f(x_0) + g(x_0, u_0) = 0 \right\} \quad (\text{V.3})$$

### V.2.3 Linéarisation locale

Cette linéarisation est effectuée au voisinage d'un point d'équilibre (équation (V.3)) selon une approximation au premier ordre des équations du système. Pour se faire, on développe les champs de vecteur  $f$  et  $g$  en série de Taylor à l'ordre 1 au voisinage de ce point d'équilibre  $(x_0, u_0)$ .

On obtient alors :

$$f(x) + g(x, u) = f(x_0) + g(x_0, u_0) + \left. \frac{\partial(f+g)}{\partial x} \right|_{(x_0, u_0)} (x - x_0) + \left. \frac{\partial g}{\partial u} \right|_{(x_0, u_0)} (u - u_0) + o^2(x - x_0, u - u_0) \quad (\text{V.4})$$

Puisque  $(x_0, u_0)$  est un point d'équilibre ( $f(x_0)+g(x_0, u_0)=0$ ) et en négligeant les termes d'ordre supérieurs à 1, on peut obtenir une approximation linéaire du modèle au premier ordre en considérant les variables d'écart :

$$\delta \ddot{x} = \left. \frac{\partial(f+g)}{\partial x} \right|_{(x_0, u_0)} \delta x + \left. \frac{\partial g}{\partial u} \right|_{(x_0, u_0)} \delta u \quad (\text{V.5})$$

Avec :  $\delta x = x - x_0$  et  $\delta u = u - u_0$ .

### V.2.4 Le modèle linéaire tangent

De la linéarisation locale (V.5), on définit le modèle linéaire tangent comme le modèle linéaire relatif aux points d'équilibre du système correspondants donc à l'ensemble des modèles linéarisés locaux.

Ce modèle linéaire tangent est ainsi défini par un ensemble de systèmes d'équations linéaires de la forme :

$$\delta \dot{x} = A(x_0, u_0) \cdot \delta x + B(x_0, u_0) \cdot \delta u \quad (\text{V.6})$$

$$\text{Où } A(x_0, u_0) = \left. \frac{\partial(f+g)}{\partial x} \right|_{(x_0, u_0)} \quad \text{et } B(x_0, u_0) = \left. \frac{\partial g}{\partial u} \right|_{(x_0, u_0)} \quad / (x_0, u_0) \in \mathcal{E}$$

Notre système a été linéarisé autour de 9 points d'équilibres selon les angles d'élévations et d'azimut. Le tableau suivant définit leur ordre (afin qu'on puisse les repérer facilement par la suite) ainsi que les commandes  $(u_v, u_h)$  correspondantes:

**Tableau V.1 :** points de fonctionnement choisis pour la linéarisation

$\alpha_h \backslash \alpha_v$	$-\pi/6$	0	$\pi/6$
$\pi/6$	1) (0.4748,0)	2) (0.7832,0)	3) (0.9790,0)
0	4) (0.4748,0)	5) (0.7832,0)	6) (0.9790,0)
$-\pi/6$	7) (0.4748,0)	8) (0.7832,0)	9) (0.9790,0)

On obtient donc les différentes matrices définissant les modèles linéaires correspondant aux différents points d'équilibre cités dans le tableau (V.1).

$$\begin{aligned}
 A_1 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.1787 & -0.0977 & 0.1655 & 0 & -0.2319 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ 0.1573 & 0 & 0.3032 & 0 & 22.6999 & 0 \\ -0.0015 & 0 & -0.0029 & 0 & -0.2156 & 0.1736 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} & A_2 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.1153 & -0.0977 & 0.1907 & 0 & 0 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ 0 & 0 & 0.1802 & 0 & 17.8214 & 0 \\ 0 & 0 & -0.0017 & 0 & -0.1693 & 0.2004 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} & A_3 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.0243 & -0.0977 & 0.1708 & 0 & 0.2122 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ -0.2167 & 0 & 0.1521 & 0 & 22.6999 & 0 \\ 0.0021 & 0 & -0.0014 & 0 & -0.2156 & 0.1736 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} \\
 A_4 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.1787 & -0.0977 & 0.1655 & 0 & -0.2319 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ 0.1573 & 0 & 0.3032 & 0 & 22.6999 & 0 \\ -0.0015 & 0 & -0.0029 & 0 & -0.2156 & 0.1736 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} & A_5 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.1153 & -0.0977 & 0.1907 & 0 & 0 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ 0 & 0 & 0.1802 & 0 & 17.8214 & 0 \\ 0 & 0 & -0.0017 & 0 & -0.1693 & 0.2004 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} & A_6 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.0243 & -0.0977 & 0.1708 & 0 & 0.2122 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ -0.2167 & 0 & 0.1521 & 0 & 22.6999 & 0 \\ 0.0021 & 0 & -0.0014 & 0 & -0.2156 & 0.1736 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} \\
 A_7 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.1787 & -0.0977 & 0.1655 & 0 & -0.2319 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ 0.1573 & 0 & 0.3032 & 0 & 22.6999 & 0 \\ -0.0015 & 0 & -0.0029 & 0 & -0.2156 & 0.1736 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} & A_8 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.1153 & -0.0977 & 0.1907 & 0 & 0 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ 0 & 0 & 0.1802 & 0 & 17.8214 & 0 \\ 0 & 0 & -0.0017 & 0 & -0.1693 & 0.2004 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix} & A_9 &= \begin{bmatrix} 0 & 17.9064 & 0 & 0 & 0 & 3.5158 \\ -0.0243 & -0.0977 & 0.1708 & 0 & 0.2122 & -0.0192 \\ 0 & 0 & -0.6983 & 0 & 0 & 0 \\ -0.2167 & 0 & 0.1521 & 0 & 22.6999 & 0 \\ 0.0021 & -0.0014 & 0 & 0 & -0.2156 & 0.1736 \\ 0 & 0 & 0 & 0 & 0 & -2.6028 \end{bmatrix}
 \end{aligned}$$

Le champ de vecteur  $g$  ne dépendant pas de  $x$ , la valeur de la matrice  $B$  reste la même pour tous les points de fonctionnement :

$$B_i = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.6983 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 2.6028 \end{bmatrix} \quad \text{avec } i=1 : 9.$$

$$\text{La matrice } C_i = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \quad i=1 : 9$$

### V.3 Commandabilité du simulateur

La commandabilité d'un système non linéaire passe par le calcul du rang de la matrice commandabilité  $Co$ . Celle-ci se calcule via les crochets de Lie jusqu'à l'ordre  $n-1$  :

$$Co = \begin{bmatrix} g(x) & -ad_f g(x) & ad_f^2 g(x) & -ad_f^3 g(x) & ad_f^4 g(x) & -ad_f^5 g(x) \end{bmatrix}$$

Avec :

$$ad_f g(x) = [f, g] = \frac{\partial g}{\partial x} \cdot f - \frac{\partial f}{\partial x} \cdot g$$

$$\text{Et } \begin{cases} ad_f^k = [f, ad_f^{k-1} g(x)] \\ ad_f^0 g(x) = g(x) \end{cases}$$

Si le rang cette matrice commandabilité est complet ( $\text{rang}(Co)=n$ ), on conclut que le système est faiblement commandable. Mais ce calcul étant fastidieux, on peut passer par le calcul des différentes commandabilités des modèles linéaires, cela conduit à la commandabilité locale.

On définit les matrices de commandabilité linéaire par :

$$.Co_i = \begin{bmatrix} B & A_i B & A_i^2 B & A_i^3 B & A_i^4 B & A_i^5 B \end{bmatrix} \quad \text{Avec : } i=1 : 9$$

Le rang de ces matrices est de 6, on conclut donc que le système est localement faiblement commandable. On peut donc procéder au calcul des différents régulateurs.

## V.4 Commande multi contrôleurs avec commutateur à événements discrets

### V.4.1 Synthèse du commutateur à événements discret

Le commutateur à événements discrets (CED) se compose en deux parties : le générateur d'événements qui permet de positionner la sortie du système dans un intervalle donné, et l'automate qui se charge de sélectionner le contrôleur correspondant à cet intervalle d'après un GRAFCET .

La sélection du contrôleur à appliquer parmi les N contrôleurs se base sur les deux sorties du TRMS (angle d'élévation et angle d'azimut).

On définit  $\varepsilon > 0$  comme étant la distance qui sépare deux points de fonctionnement. Pour le TRMS trois points de fonctionnement ont été choisis pour chacun des deux mouvements (horizontal et vertical). A chaque point de fonctionnement sera associé une zone de fonctionnement comme suit:

Pour l'angle d'élévation:

$$R_{v1} \text{ correspond à l'intervalle : } \left[ \alpha_{v0}(1) - \frac{\varepsilon}{2}, \alpha_{v0}(1) + \frac{\varepsilon}{2} \right]; \text{ avec } \alpha_{v0}(1) = -\frac{\pi}{6}$$

$$R_{v2} \text{ correspond à l'intervalle : } \left[ \alpha_{v0}(2) - \frac{\varepsilon}{2}, \alpha_{v0}(2) + \frac{\varepsilon}{2} \right]; \text{ avec } \alpha_{v0}(2) = 0$$

$$R_{v3} \text{ correspond à l'intervalle : } \left[ \alpha_{v0}(3) - \frac{\varepsilon}{2}, \alpha_{v0}(3) + \frac{\varepsilon}{2} \right]; \text{ avec } \alpha_{v0}(3) = \frac{\pi}{6}$$

Pour l'angle d'azimut :

$R_{h1}$  correspond à l'intervalle :  $[\alpha_{h0}(1) - \frac{\varepsilon}{2}, \alpha_{h0}(1) + \frac{\varepsilon}{2}]$  ; avec  $\alpha_{h0}(1) = \frac{\pi}{6}$

$R_{h2}$  correspond à l'intervalle :  $[\alpha_{h0}(2) - \frac{\varepsilon}{2}, \alpha_{h0}(2) + \frac{\varepsilon}{2}]$  ; avec  $\alpha_{h0}(2) = 0$

$R_{h3}$  correspond à l'intervalle :  $[\alpha_{h0}(3) - \frac{\varepsilon}{2}, \alpha_{h0}(3) + \frac{\varepsilon}{2}]$  ; avec  $\alpha_{h0}(3) = -\frac{\pi}{6}$

Le générateur d'événements contient les 9 règles suivantes :

Si  $\alpha_V \in R_{v1}$  et  $\alpha_h \in R_{h1}$  alors  $P_1 = 1$ ;

Si  $\alpha_V \in R_{v1}$  et  $\alpha_h \in R_{h2}$  alors  $P_2 = 1$ ;

Si  $\alpha_V \in R_{v1}$  et  $\alpha_h \in R_{h3}$  alors  $P_3 = 1$ ;

Si  $\alpha_V \in R_{v2}$  et  $\alpha_h \in R_{h1}$  alors  $P_4 = 1$ ;

Si  $\alpha_V \in R_{v2}$  et  $\alpha_h \in R_{h2}$  alors  $P_5 = 1$ ;

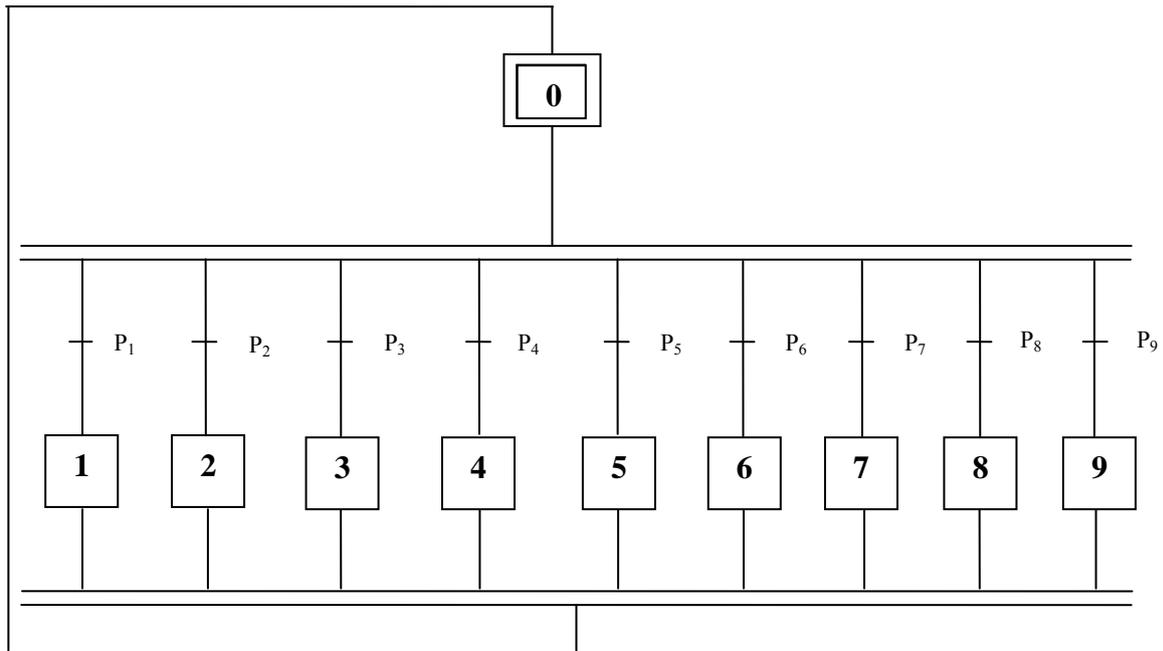
Si  $\alpha_V \in R_{v2}$  et  $\alpha_h \in R_{h3}$  alors  $P_6 = 1$ ;

Si  $\alpha_V \in R_{v3}$  et  $\alpha_h \in R_{h1}$  alors  $P_7 = 1$ ;

Si  $\alpha_V \in R_{v3}$  et  $\alpha_h \in R_{h2}$  alors  $P_8 = 1$ ;

Si  $\alpha_V \in R_{v3}$  et  $\alpha_h \in R_{h3}$  alors  $P_9 = 1$ ;

Après avoir déterminé les intervalles de fonctionnement, l'étape suivante consiste à exécuter le GRAFCET représenté sur la figure (V.1) afin d'activer le contrôleur convenable à la configuration du procédé :



**Figure V.1 :** *Grafset du commutateur*

Tel que :

- L'étape 1 correspond à l'activation du régulateur 1 ;
- L'étape 2 correspond à l'activation du régulateur 2 ;
- L'étape 3 correspond à l'activation du régulateur 3 ;
- L'étape 4 correspond à l'activation du régulateur 4 ;
- L'étape 5 correspond à l'activation du régulateur 5 ;
- L'étape 6 correspond à l'activation du régulateur 6 ;
- L'étape 7 correspond à l'activation du régulateur 7 ;
- L'étape 8 correspond à l'activation du régulateur 8 ;
- L'étape 9 correspond à l'activation du régulateur 9.

La figure (V.2) décrit le schéma d'implémentation sous *Stateflow* de *Matlab* :

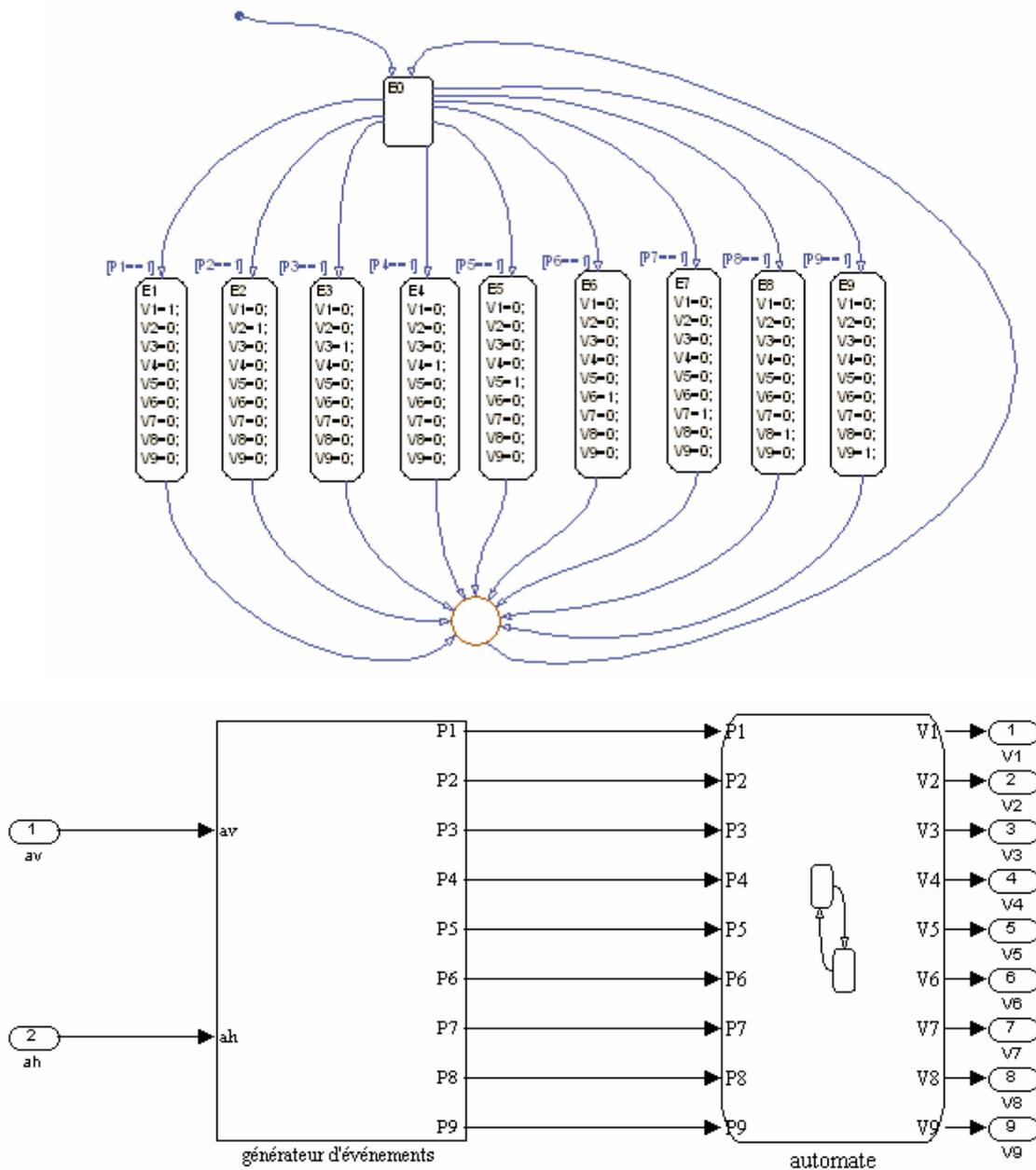


Figure V.2 : schéma d'implémentation sous stateflow

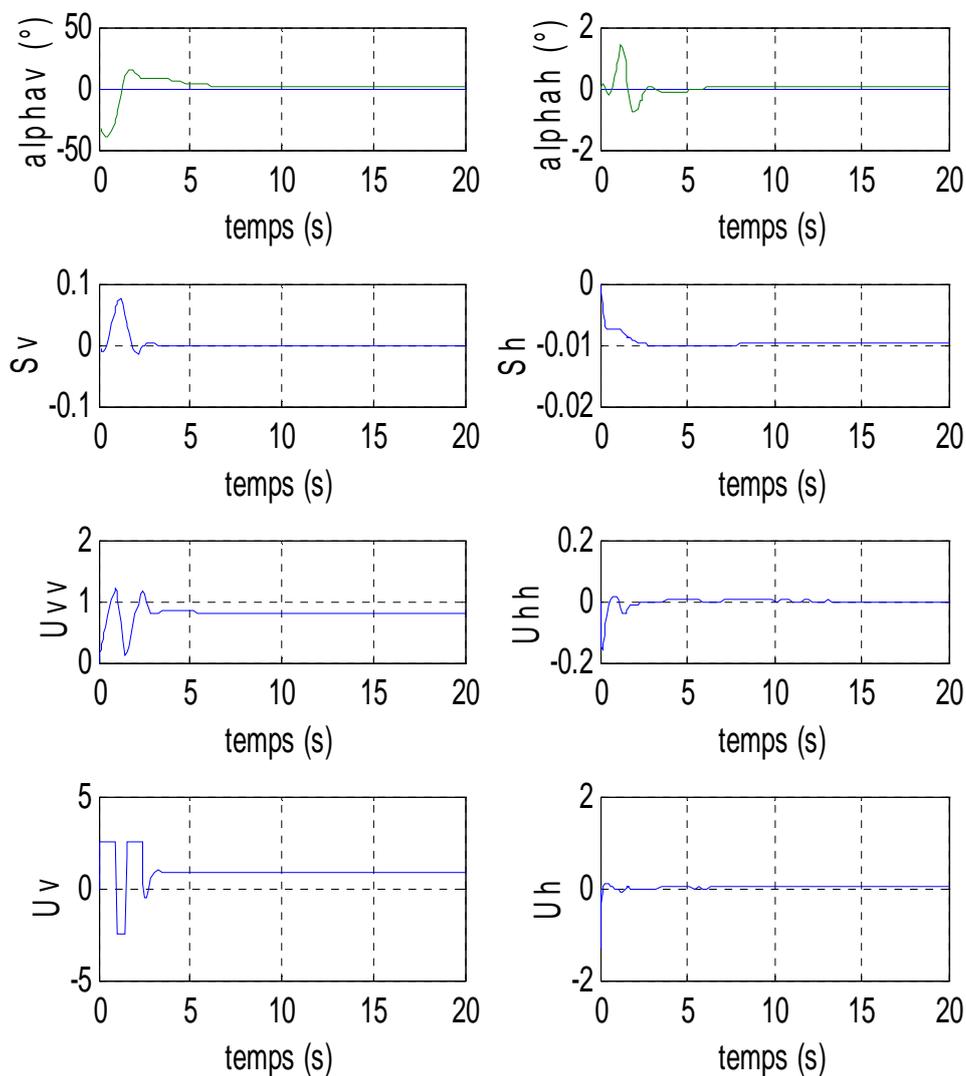
Les  $V_i, i=1,2,\dots,9$ , sont les fonctions d'activation des différents régulateurs, qui prennent leurs valeurs dans  $\{0, 1\}$ .

### V.4.2 Commande multi contrôleurs par retour d'état

La première commande à appliquer au simulateur d'hélicoptère est la commande par retour d'état en utilisant les matrices calculées dans la section (V.2) .

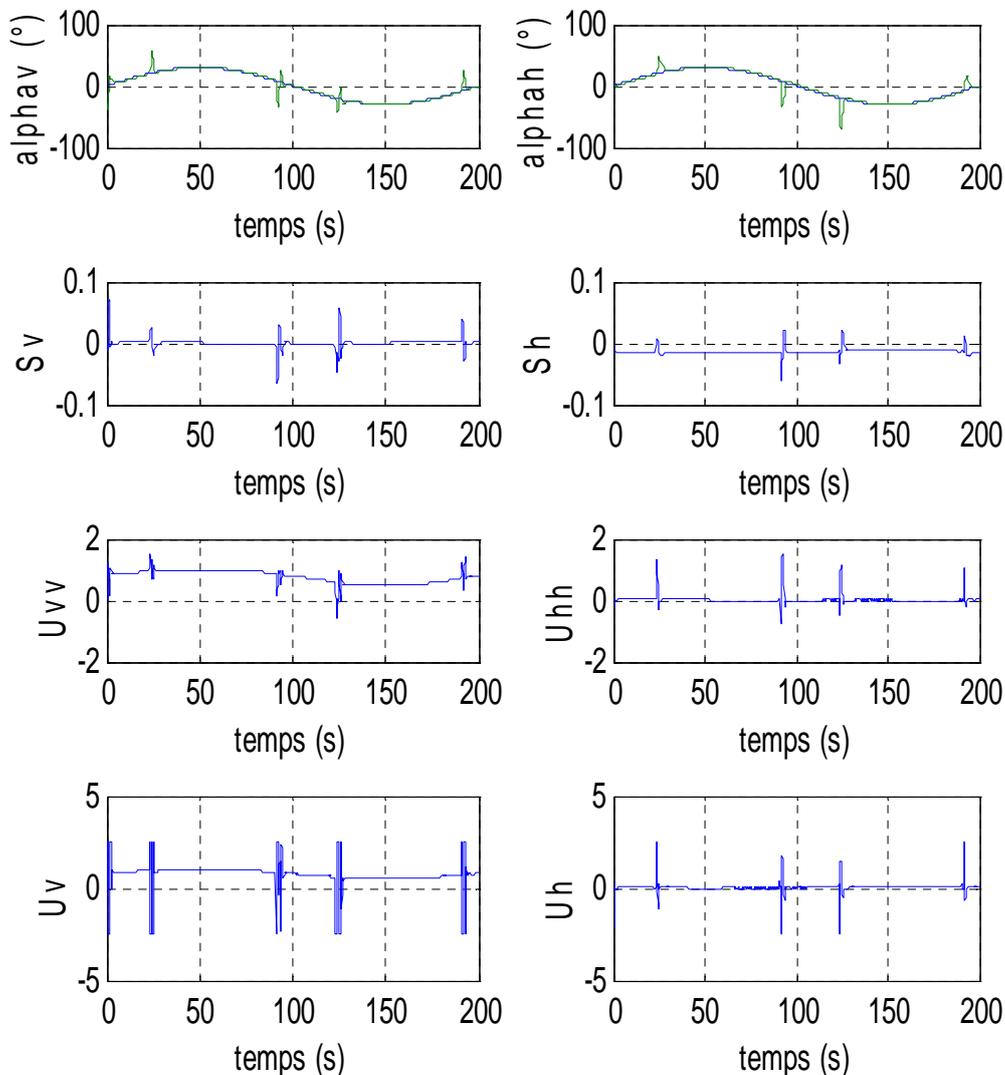
Le placement de pôles a été fait en  $[-7, -0.5, -5, -7, -0.5, -5, -4, -4]$ , les deux derniers pôles correspondent aux gains des deux actions intégrales ajoutées dans le but d'annuler l'erreur statique due aux perturbations.

La figure suivante montre les réponses du système pour une référence nulle (Stabilisation en  $(0,0)$ ) :



**Figure V.3 :** réponses du système et signaux de commande pour une stabilisation en  $(0,0)$  avec une commande par retour d'état

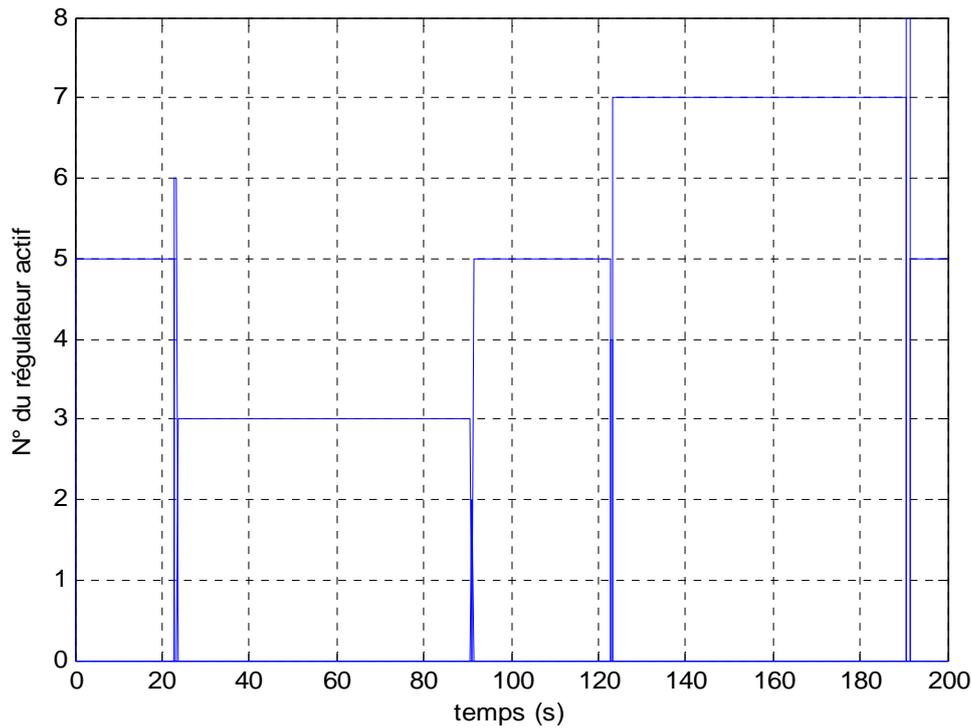
La figure ci-dessus montre les réponses pour une référence sinusoïdale d'amplitude  $\frac{\pi}{6}$  et de fréquence 0.005 Hz, ainsi que les commutations des contrôleurs :



**Figure V.4 :** réponses du système et signaux de commande pour une référence sinusoïdale avec une commande multi contrôleurs par retour d'état à commutation franche

On constate que les réponses du système présentent des sauts dus aux discontinuités au niveau des signaux de commande, ces derniers sont causés par les commutations franches d'un contrôleur à un autre (figure V.5).

Ce problème sera résolu dans la suite de ce chapitre à l'aide de la commutation floue, qui permet d'avoir un passage progressif d'un régulateur à un autre.



**Figure V.5 :** commutation en tout ou rien des régulateurs par retour d'état pour un signal de référence sinusoïdal

#### V.4.3 Commande multi contrôleurs $H_\infty$

La synthèse d'une commande  $H_\infty$  passe d'abord par la détermination des fonctions de pondération  $W_1(s)$  et  $W_2(s)$ . Ceci a été décrit dans le chapitre précédent.

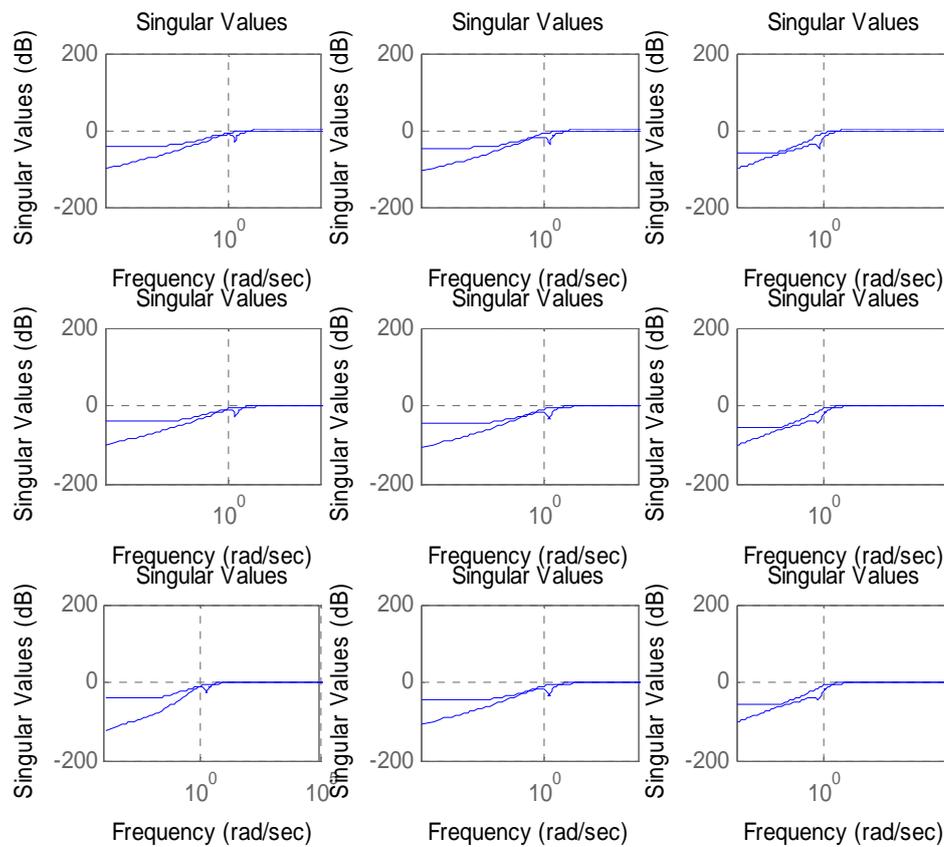
Le simulateur d'hélicoptère étant un système multi variable à deux entrées et deux sorties, on prend des matrices de fonctions de pondération. Le calcul des régulateurs se fait à l'aide de la fonction Matlab *ncfsyn* qui permet de faire une synthèse centralisée.

Après plusieurs simulations, le choix s'est porté sur les pondérations suivantes relatives à tous les points de fonctionnement :

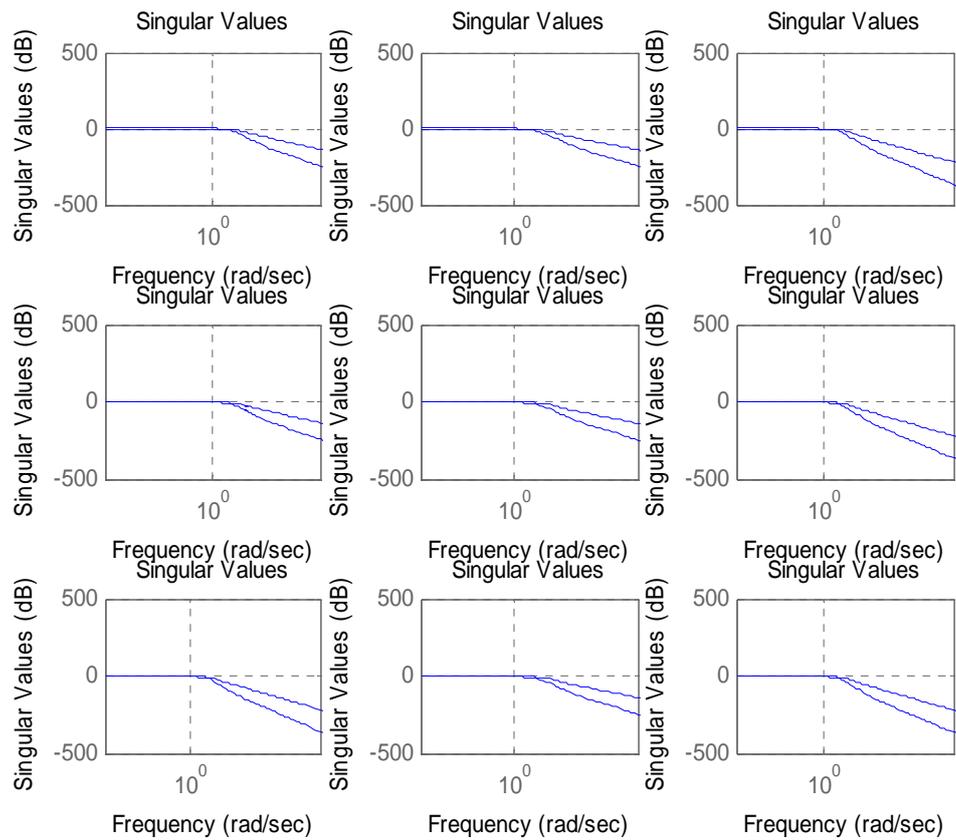
$$W_1(s) = \begin{bmatrix} \frac{0.8s + 0.4}{s + 0.02} & 0 \\ 0 & \frac{0.8s + 0.05}{s + 0.02} \end{bmatrix} \quad (\text{V.7})$$

$$W_2(s) = \begin{bmatrix} 30 & 0 \\ 0 & 16 \end{bmatrix} \quad (\text{V.8})$$

La fonction *ncfsyn* de Matlab, permet d'obtenir des régulateurs dont l'ordre est égal à l'ordre du système augmenté des fonctions de pondération, pour le TRMS les correcteurs sont d'ordre 17. Pour les implémenter on est amené à faire une réduction d'ordre à l'aide de la commande *strunc* de Matlab, on obtient ainsi des régulateurs du 8<sup>ème</sup> ordre facile à implémenter. Les figures (V.6) et (V.7) représentent le tracé des valeurs singulières des matrices de fonctions de sensibilité et de sensibilité complémentaire du système avec le correcteur en boucle fermée pour les différents points de fonctionnement choisis pour la synthèse :

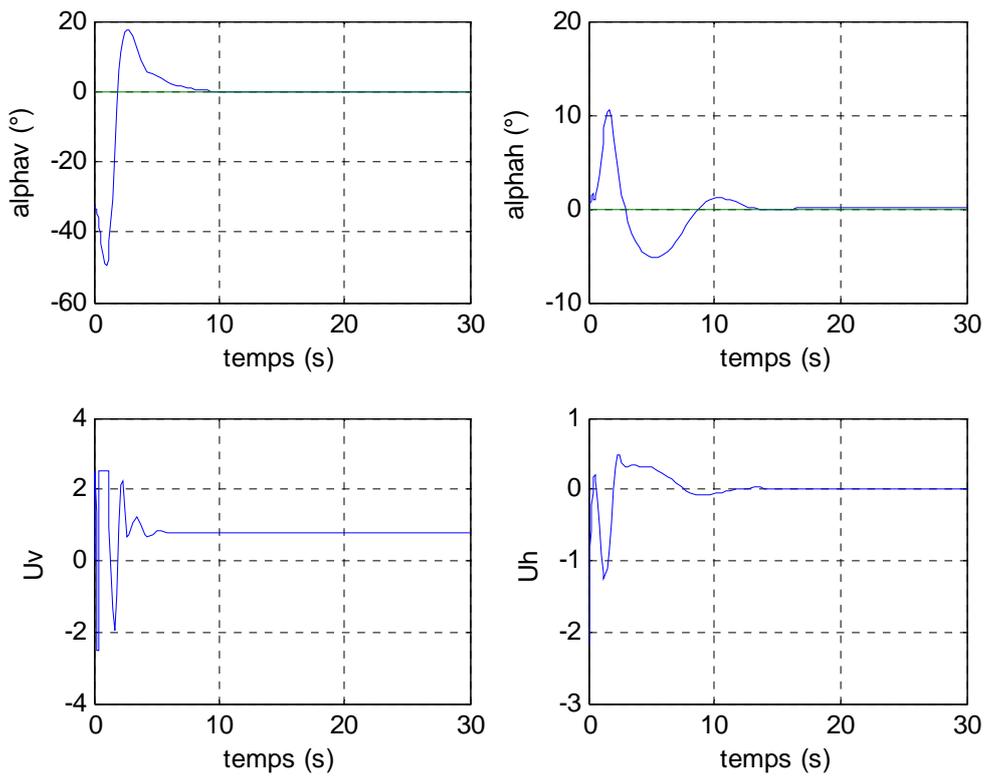


**Figure V.6 :** tracé des valeurs singulières des matrices fonctions de sensibilité des différents points de fonctionnement



**Figure V.7 :** tracé des valeurs singulières des matrices fonctions de sensibilité complémentaires des différents points de fonctionnement.

Les réponses du système bouclé pour une stabilisation en (0,0) sont données ci-dessous :



**Figure V.8 :** réponses du système et signaux de commande pour une stabilisation en  $(0,0)$  avec une commande  $H_\infty$

Et la figure V.9, illustre les réponses du procédé à référence en créneaux d'amplitude  $30^\circ$ , ainsi que la commutation des différents contrôleurs :

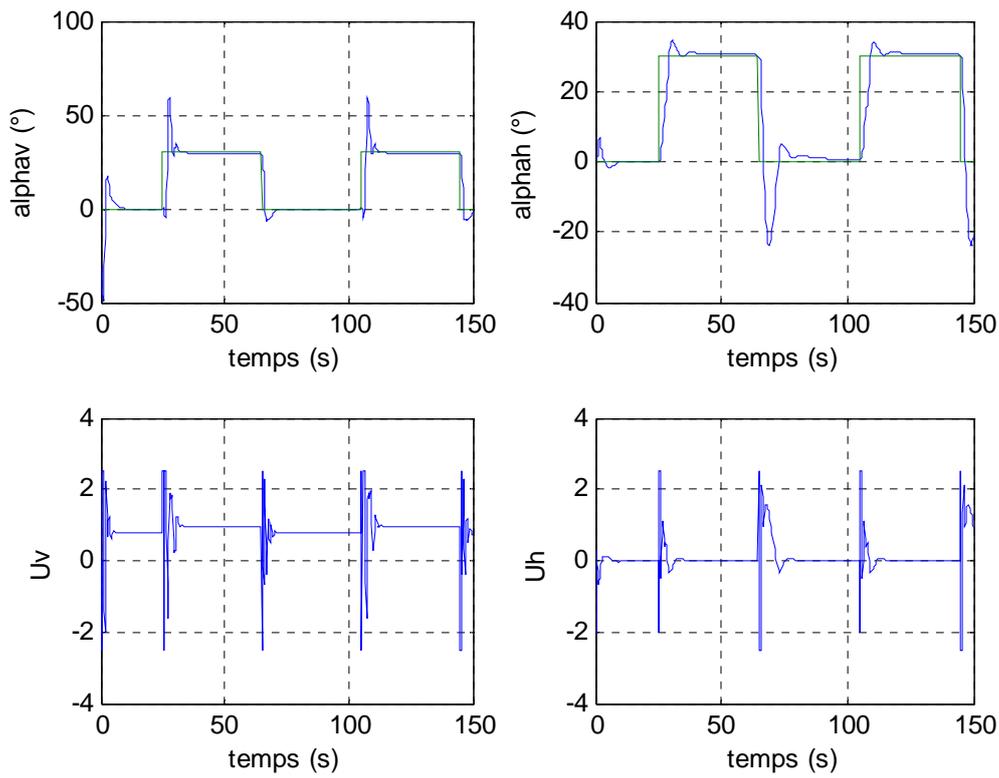


Figure V.9 : réponses du système et signaux de commande pour un signal de référence carré avec une commande multi contrôleurs  $H^\infty$  à commutation franche.

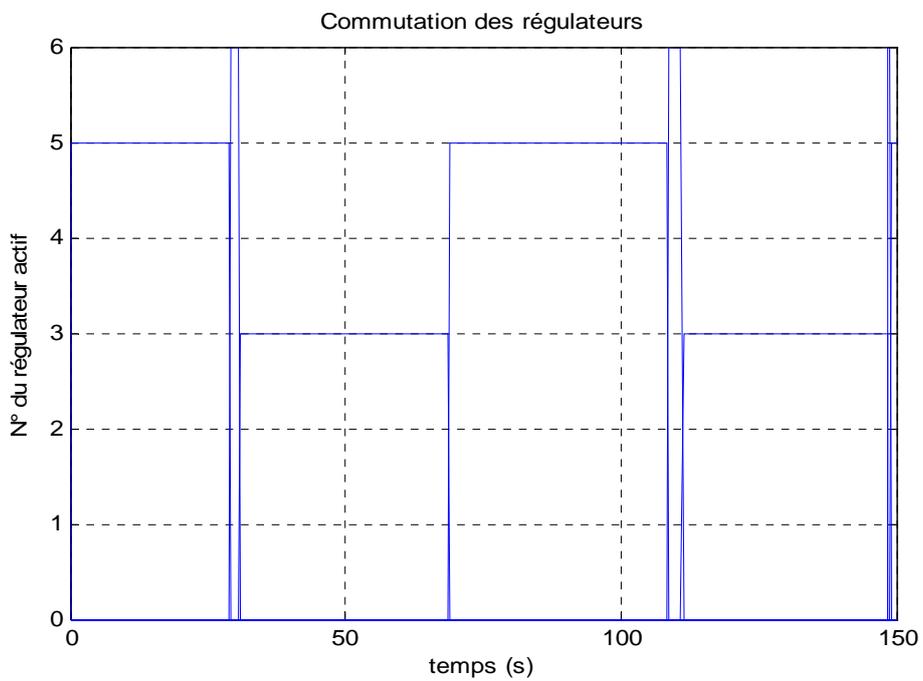


Figure V.10 : commutation en tout ou rien des régulateurs  $H^\infty$  pour un signal de référence carré.

Poursuite d'une sinusoïde d'amplitude  $30^\circ$  et de fréquence 0.02 Hz :

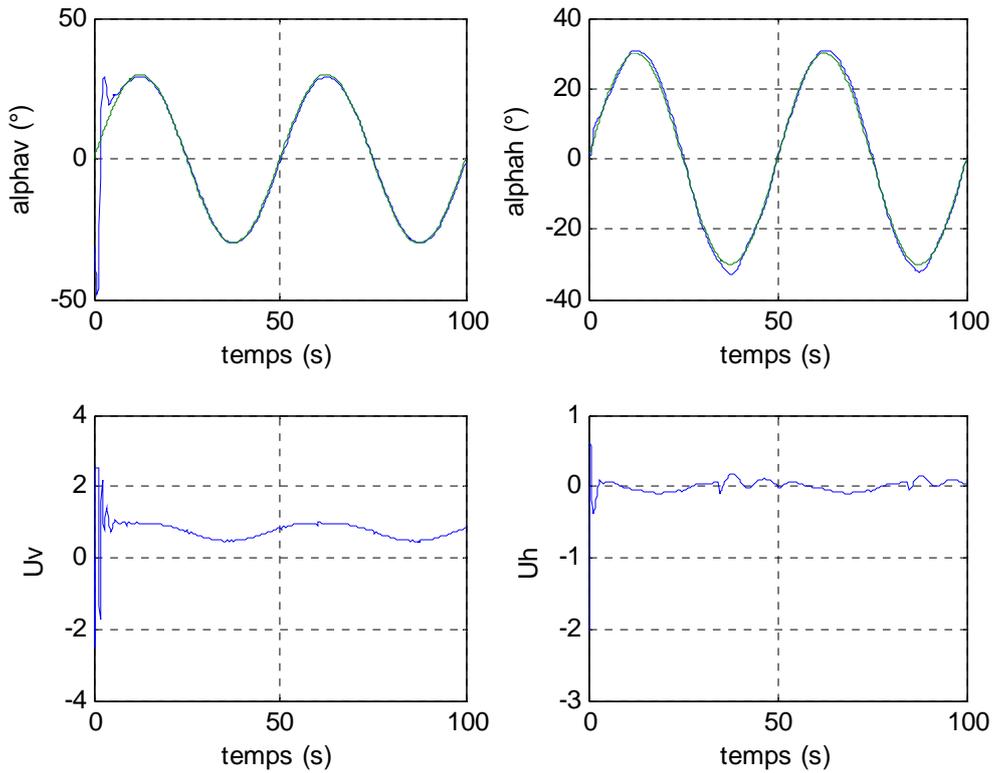


Figure V.11 : réponses du système et signaux de commande pour un signal de référence en sinusoïde en référence avec une commande multi contrôleurs  $H^\infty$  à commutation franche.

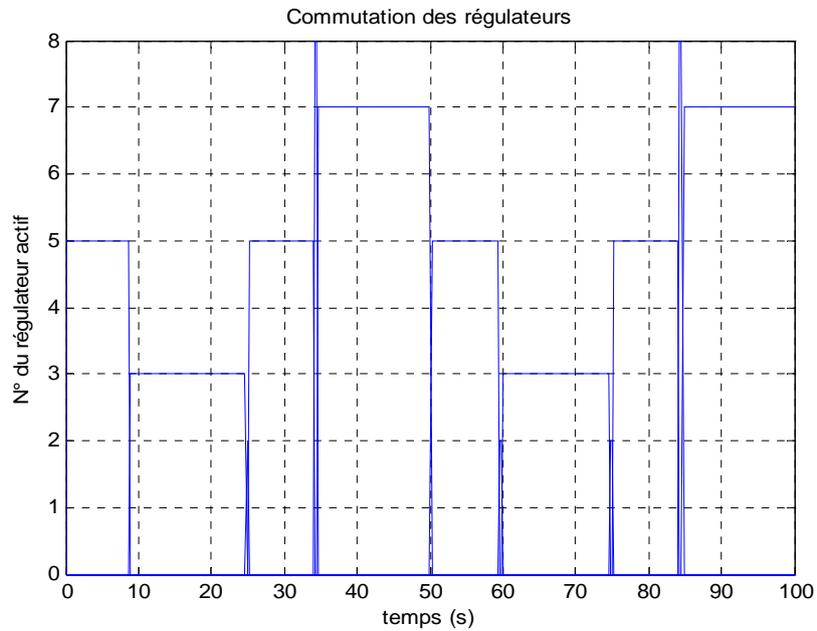
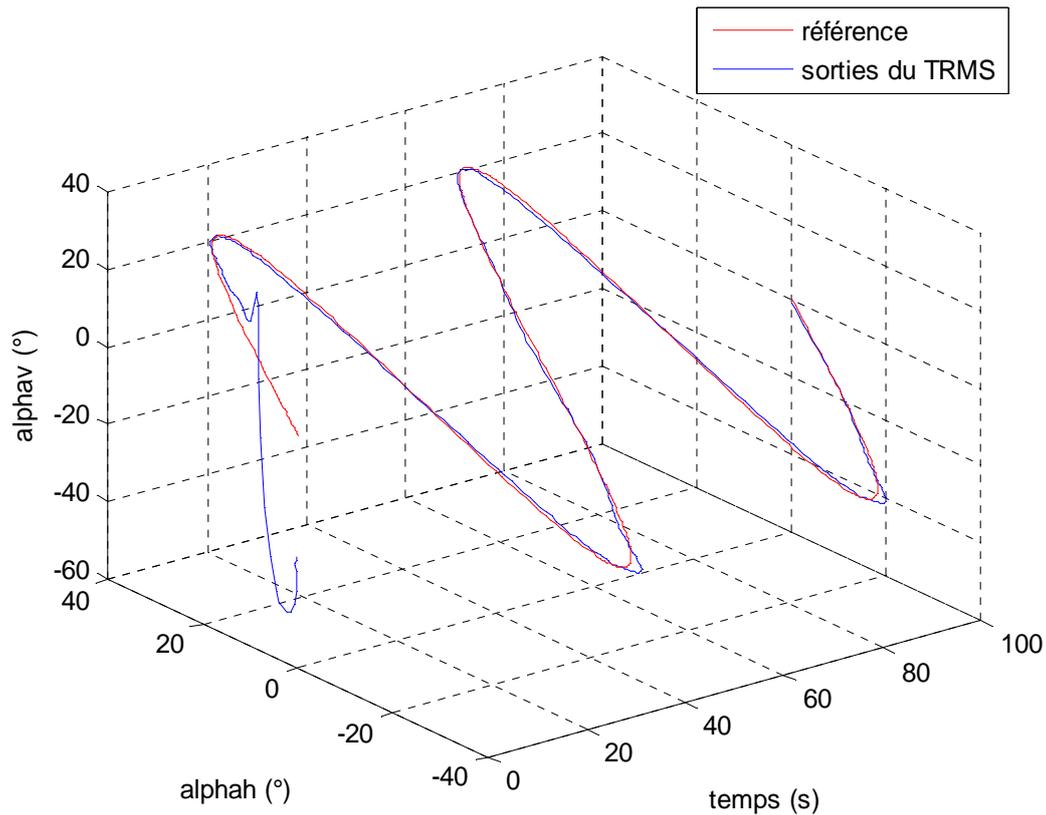


Figure V.12 : commutation en tout ou rien des régulateurs  $H^\infty$  pour un signal de référence en sinusoïde



**Figure V.13 :** trajectoire dans l'espace d'une poursuite de sinusoides avec une commande multi contrôleurs  $H^\infty$  à commutation franche

Les résultats de simulation obtenus précédemment, ont montré l'efficacité de la commande  $H^\infty$ , car elle a pu assurer la stabilisation du système en presque 8 sec, avec des commandes admissibles par les actionneurs. De plus l'aspect multi contrôleurs nous a permis d'avoir une bonne poursuite de trajectoire sinusoïdale. Néanmoins les commutations franches, où on a un seul régulateur actif à la fois, provoquent des discontinuités dans les signaux de commande qui peuvent se traduire par des sauts dans les réponses du système. Pour résoudre ce problème, une commutation floue est proposée dans la suite de ce chapitre où le GRAFCET du commutateur à événements discret sera remplacé par un système flou du type Takagi-Sugeno.

#### V.4.4 Commande multi contrôleurs RST

Pour notre simulateur, la synthèse des différents contrôleurs RST s'effectuera en décentralisé. Cette approche consiste à séparer la dynamique du mouvement horizontal de celle du mouvement vertical. Les interactions entre ces sous systèmes seront perçues par les contrôleurs comme des perturbations à rejeter.

Avant de calculer les coefficients des différents polynômes constituant les contrôleurs RST, un, ou plusieurs, modèles de référence doivent être choisis de façon à décrire le comportement voulu de la sortie. Le modèle de référence doit avoir un degré relatif supérieur ou égal à celui du système.

On a pris des modèles de références pour l'horizontal et le vertical ayant comme pôles°:

$[-3 \ -4.5 \ -5.5 \ -7 \ -6]$ , cela assure une réponse rapide du système sans dépassement.

Pour ces contrôleurs on utilisera deux commutateurs, un pour sélectionner une commande parmi les commandes RST associés au sous système vertical, l'autre pour commuter entre les contrôleurs relatifs au sous système horizontal. Le commutateur du sous système d'azimut est le même que celui décrit plus haut, dans la section (V.4.1). Le modèle de ce sous système dépendant de l'angle vertical donne 9 modèles linéarisés associés aux points de fonctionnement choisis.

Par contre le commutateur du sous système vertical, ne dépendant que de  $\alpha_v$ , comporte uniquement 3 zones d'activation relatives à l'angle d'élévation. Ces zones sont celles déjà décrites en (V.4.1). Le générateur d'événement comporte donc les règles suivantes :

Si  $\alpha_v \in R_{v1}$  alors  $P_1 = 1$ ;

Si  $\alpha_v \in R_{v2}$  alors  $P_2 = 1$ ;

Si  $\alpha_v \in R_{v3}$  alors  $P_3 = 1$ ;

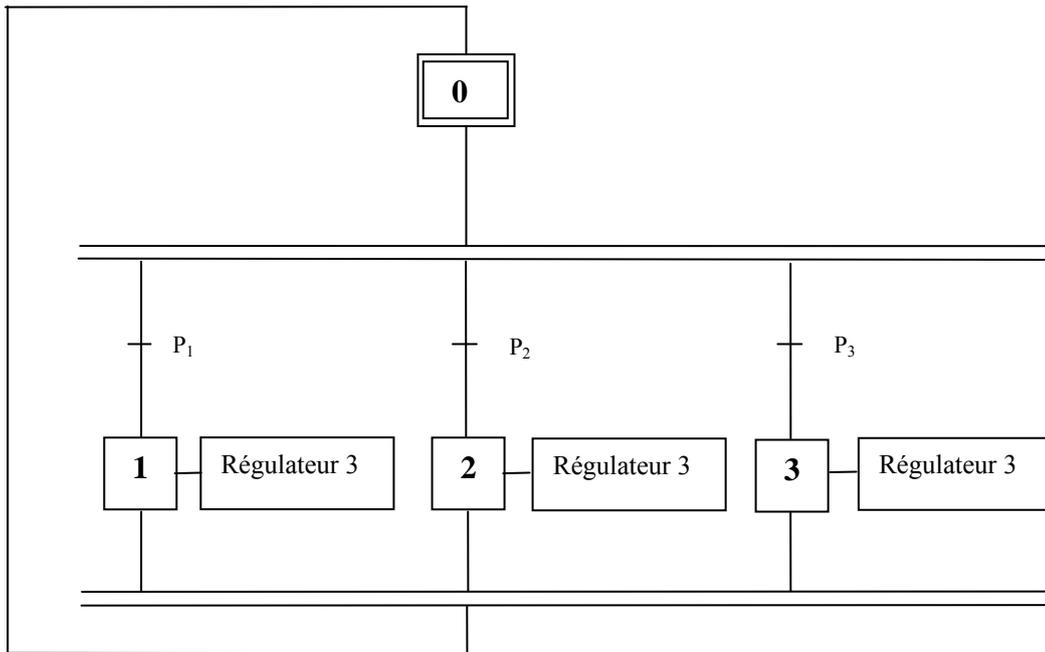


Figure V.14 : Grafcet du commutateur du sous système d'élévation.

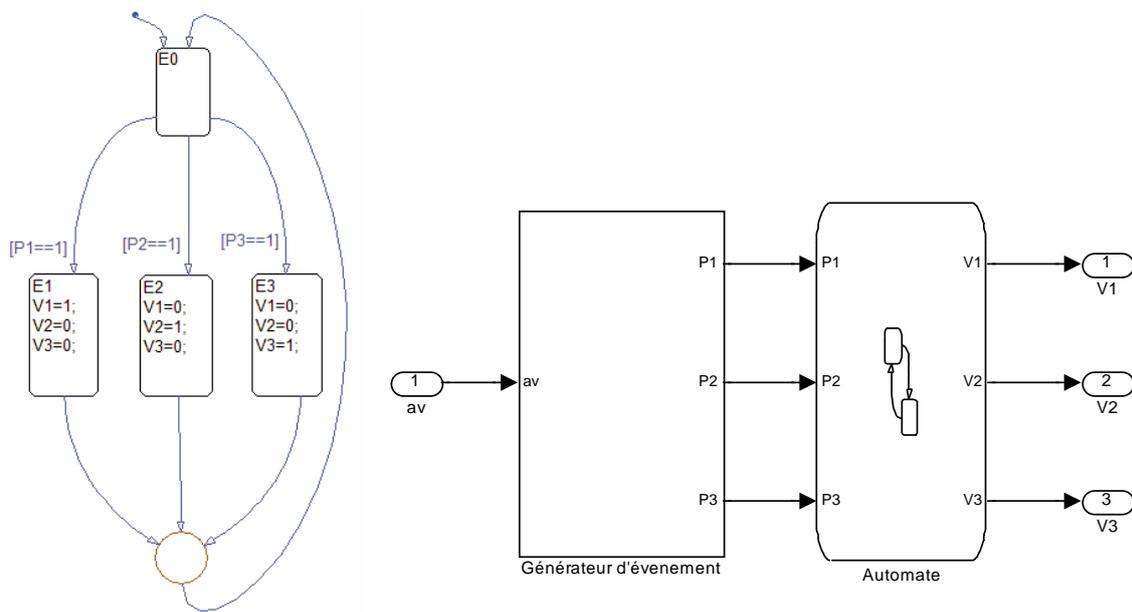


Figure V.15 : schéma d'implémentation sous stateflow du commutateur du sous système d'élévation

La figure V.16 représente les résultats de l'application des contrôleurs RST sur le système ainsi que les tensions de commandes associés pour une stabilisation en (0,0).

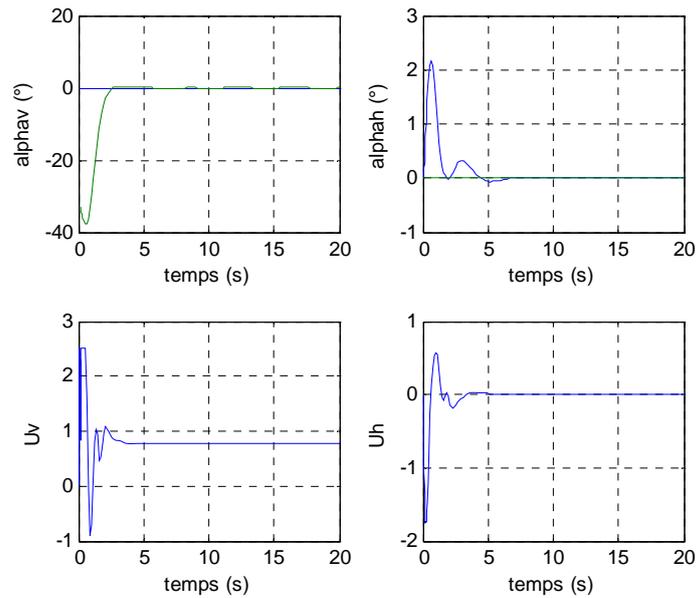


Figure V.16 : réponses du système et signaux de commande pour une stabilisation en  $(0,0)$  avec un contrôleur RST

On constate que la commande RST a réussi à stabiliser le système, tout en ayant des tensions de commande tolérables.

Les figures V.17 et V.18 illustrent la réponse du système à une référence en créneaux d'amplitude  $30^\circ$ , ainsi que la commutation des différents contrôleurs :

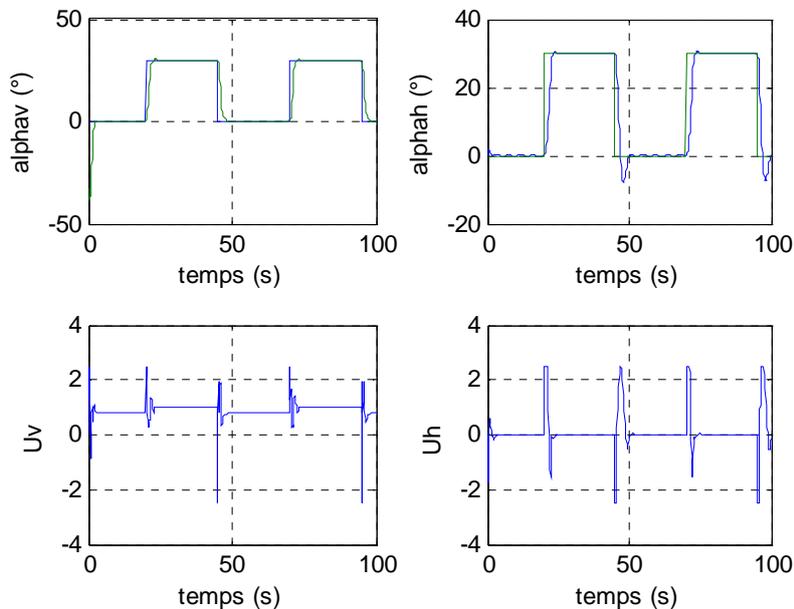


Figure V.17 : réponses du système et signaux de commande pour un signal de référence carré avec une commande multi contrôleur RST à commutation franche

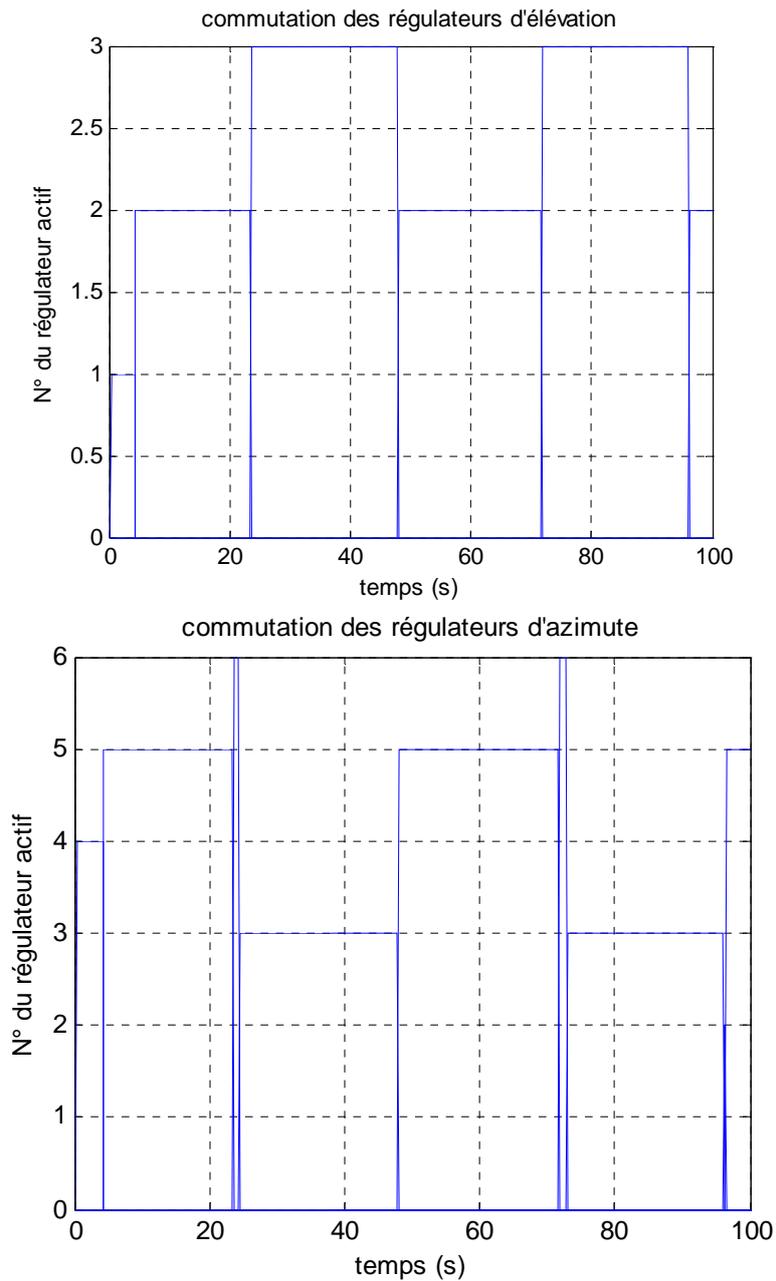
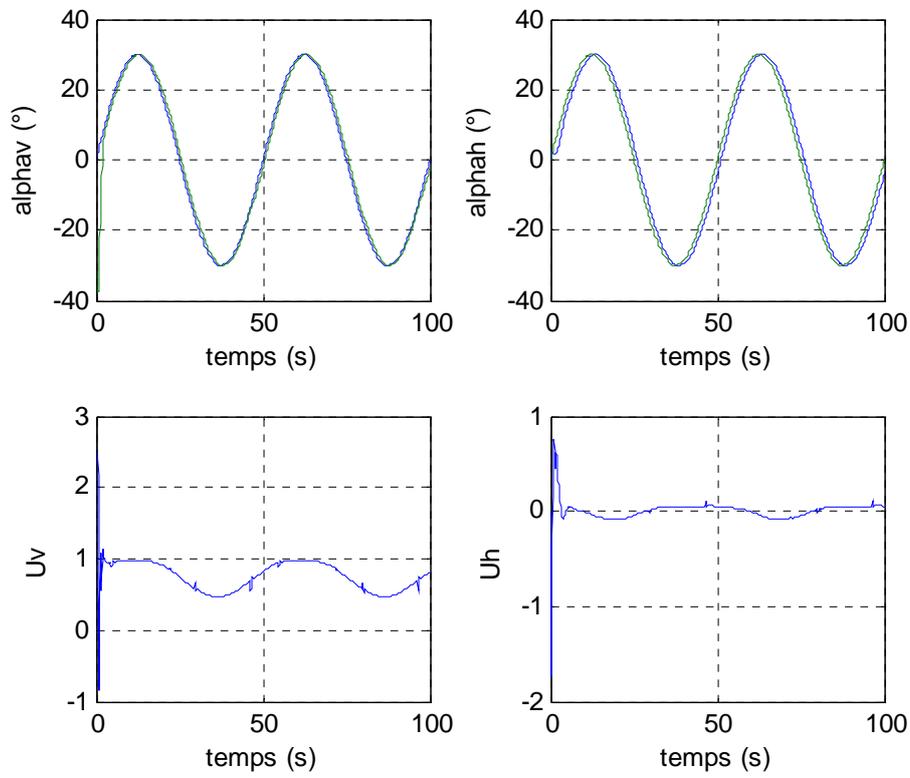


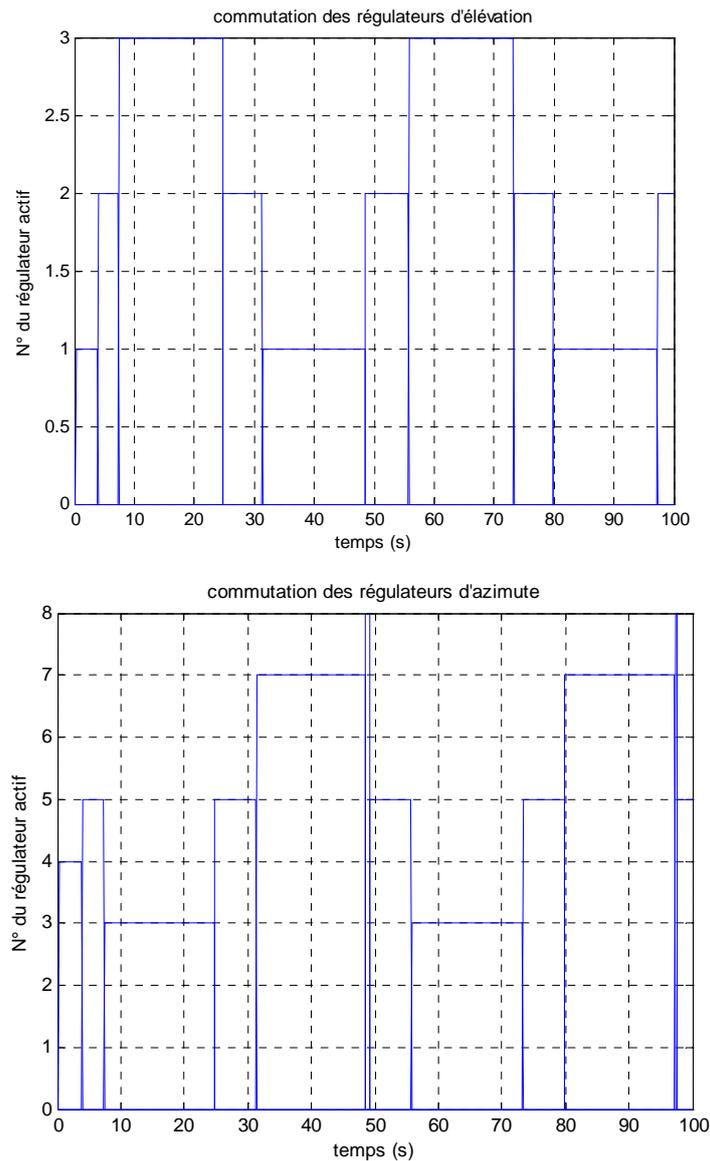
Figure V.18 : commutation en tout ou rien des régulateurs RST pour un signal de référence carré

La poursuite est satisfaisante, et les commandes sont admissibles, elles sont saturées uniquement au moment du changement de consigne.

La figure V.19 représente la réponse du système à une référence sinusoïdale d'amplitude  $30^\circ$  et de fréquence 0.02. La figure V.20 montre les régulateurs actifs à chaque instant.

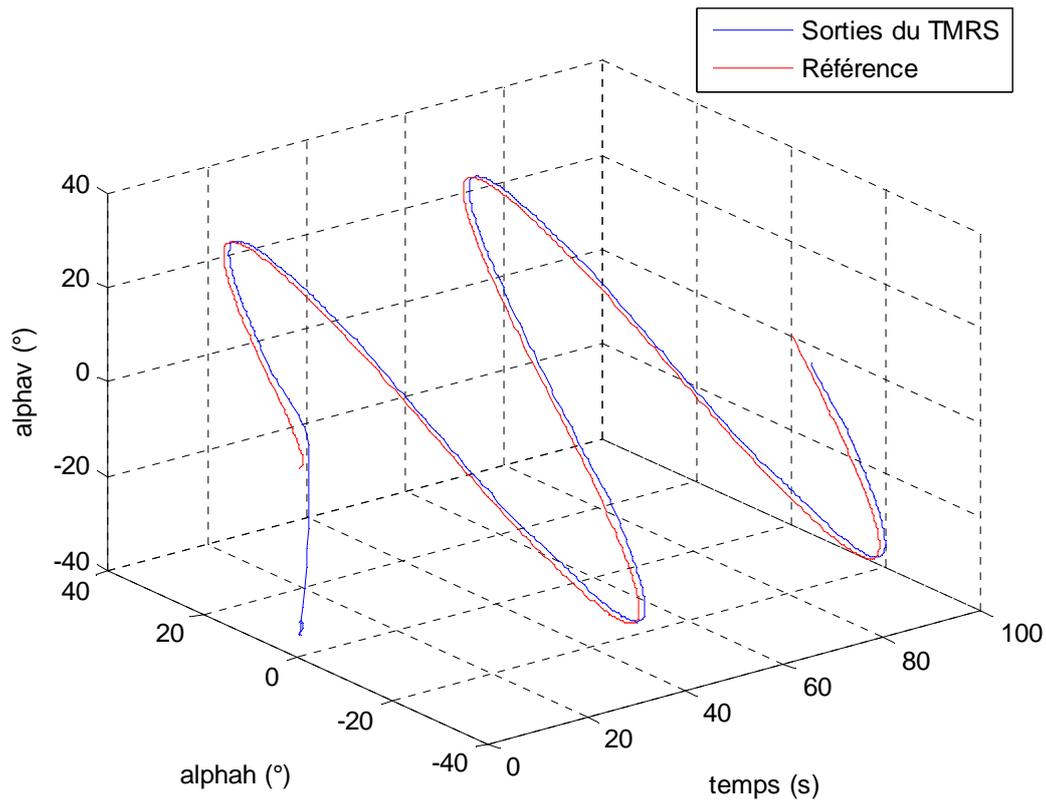


**Figure V.19 :** réponses du système et signaux de commande pour un signal de référence sinusoïdal avec une commande multi contrôleurs RST à commutation franche.



**Figure V.20 :** commutation en tout ou rien des régulateurs RST pour un signal de référence sinusoïdal

La poursuite de la sinusoïde présente une faible erreur de suivie prenant  $2^\circ$  comme valeur maximale (pour les deux sorties). On remarque aussi des discontinuités au niveau de la commande due à la commutation entre les différents contrôleurs. Ces discontinuités provoquent des sauts au niveau des sorties. Celles-ci peuvent être supprimées via la commutation floue.



**Figure V.21 :** trajectoire dans l'espace d'une poursuite de sinusoides avec une commande multi contrôleurs RST à commutation franche

## V.5 Commande multi contrôleurs avec commutateur flou

### V.5.1 Synthèse du commutateur flou

Le principe du commutateur flou est de combiner les sorties de plusieurs correcteurs locaux, afin de constituer le signal de commande final. Le commutateur devient un système flou symbolique dont les entrées sont les angles d'élévation  $\alpha_v$  et d'azimut  $\alpha_h$ . Sa sortie correspond au signal de commande à appliquer au procédé. Ainsi, la pondération des lois de commandes se fait sur la base d'un signal mesuré qui est la sortie du système.

La figure (V.22) illustre les fonctions d'appartenance qu'on a choisies pour les variables  $\alpha_v$  et  $\alpha_h$  :

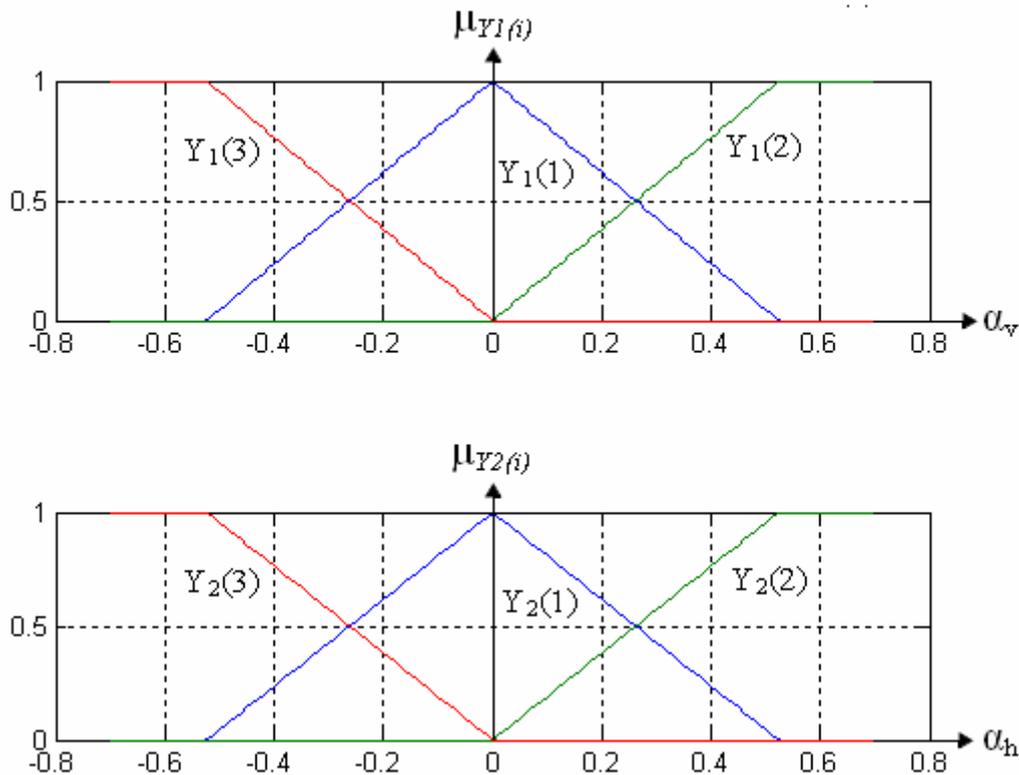


Figure V.22 : Partition floue des variables d'entrée

Les fonctions d'appartenance  $\mu_{Y_1(i)}$  et  $\mu_{Y_2(i)}$  ont une forme triangulaire, six symboles sont définis  $Y_j(i)$  et  $Y_2(i)$  avec  $i=0,1,2,3$ , tel que :

$Y_j(1)$  correspond au point de fonctionnement  $0 \text{ rad}$ ,

$Y_j(2)$  correspond au point de fonctionnement  $\frac{\pi}{6} \text{ rad}$

$Y_j(3)$  correspond au point de fonctionnement  $-\frac{\pi}{6} \text{ rad}$

Avec  $j=1,2$  (on note : 1 pour l'élévation, et 2 pour l'azimute).

Ainsi, la base des règles est obtenue à partir de la matrice de Macvicar-whelan :

**Tableau V.2 :** Base de règles, (3×3)

$\alpha_v \backslash \alpha_h$	$Y_1(3)$	$Y_1(1)$	$Y_1(2)$
$Y_2(2)$	$U_1$	$U_2$	$U_3$
$Y_2(1)$	$U_4$	$U_5$	$U_6$
$Y_2(3)$	$U_7$	$U_8$	$U_9$

Les 9 règles s'écrivent comme suit :

$$\text{Si } \alpha_v \text{ est } Y_1(i) \text{ et } \alpha_h \text{ est } Y_2(j) \text{ Alors : } U=U_l \text{ pour } l=0, \dots, 9, i=1, 2, 3 \text{ et } j=1, 2, 3 \quad (\text{V.9})$$

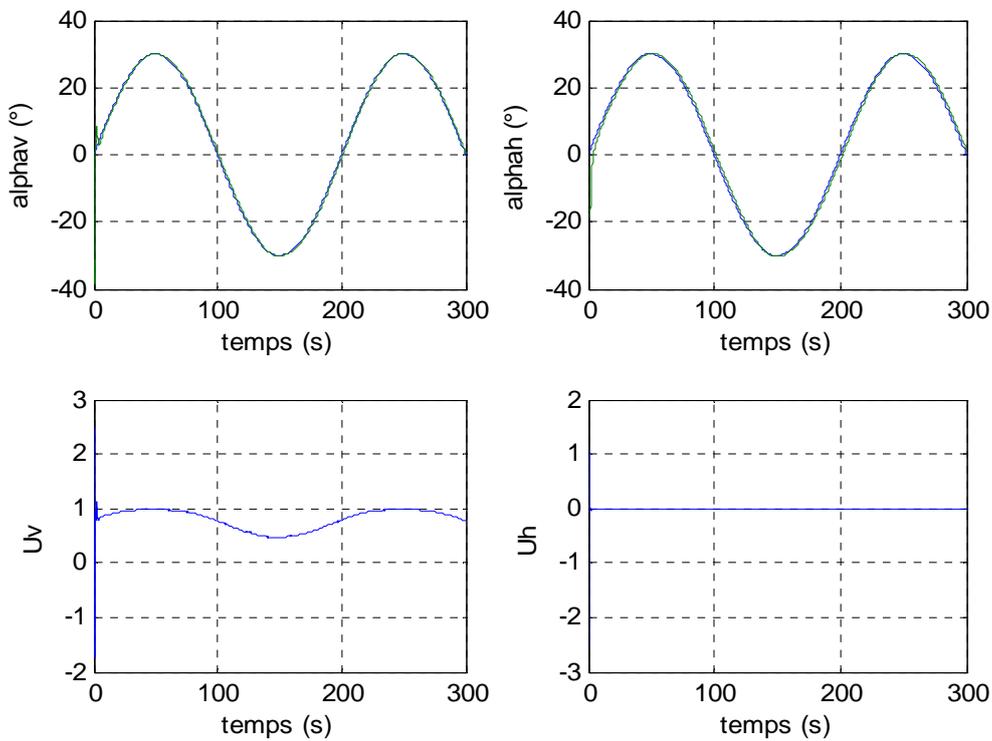
La sortie floue est inférée de la manière suivante :

$$Up = \frac{\sum_{i=1}^9 \mu_{Y_1(i) \cap Y_2(i)}(\alpha_v, \alpha_h) \times u_i}{\sum_{i=1}^9 \mu_{Y_1(i) \cap Y_2(i)}(\alpha_v, \alpha_h)} \quad (\text{V.10})$$

### V.5.2 Commande multi contrôleurs par retour d'état avec commutateur flou

Afin de vérifier si les discontinuités des signaux de commande ont bien été éliminées, les mêmes simulations effectuées pour le commutateur à événements discrets seront répétées, avec, cette fois ci, le commutateur flou synthétisé précédemment.

Poursuite de trajectoire sinusoïdale :



**Figure V.23 :** réponses du système et signaux de commande pour un signal de référence sinusoïdal avec une commande multi contrôleurs par retour d'état à commutation floue.

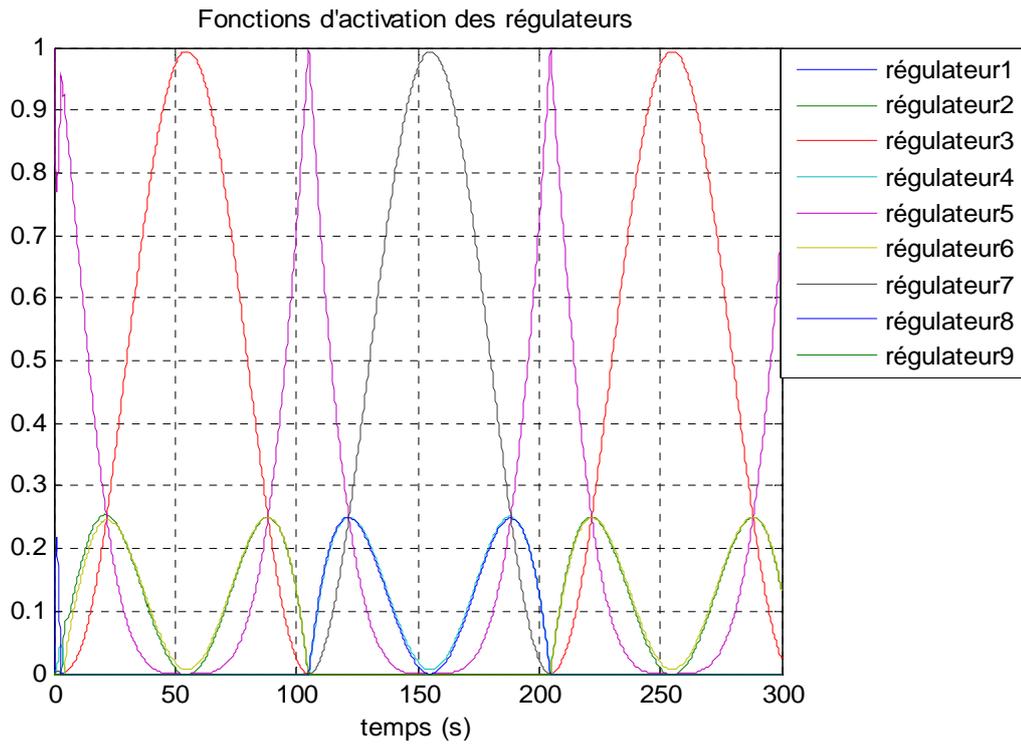
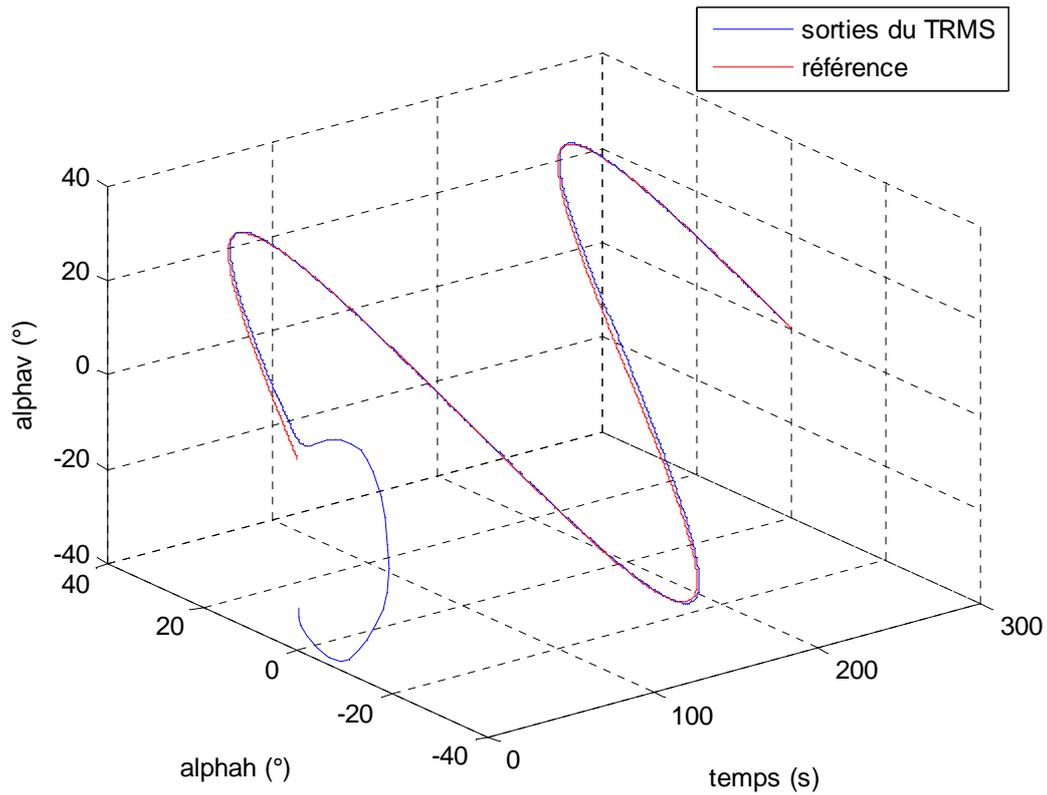


Figure V.24 : Fonctions d'activations floues des régulateurs par retour d'état pour une sinusoïde en référence.



**Figure V.25 :** trajectoire du simulateur dans l'espace avec une commande par retour d'état et commutateur flou.

On voit que les commandes sont lisses et ne présentent plus de discontinuité. La commutation floue a donc réussi à les supprimer.

Poursuite d'un signal carré :

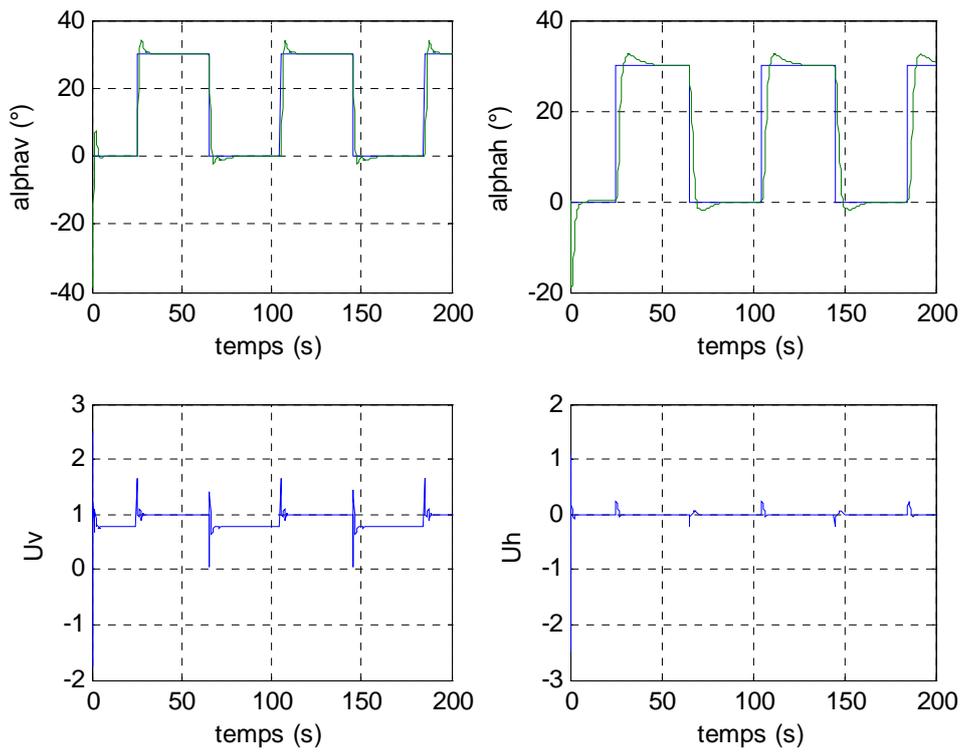


Figure V.26 : réponses du système à une référence carrée avec une commande multi contrôleurs par retour d'état à commutation floue.

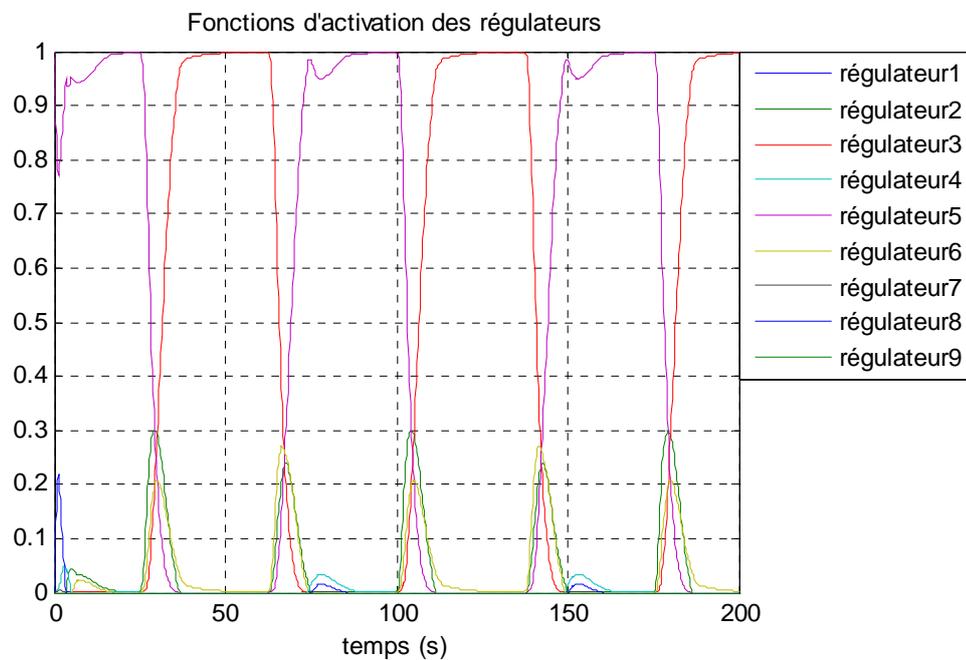
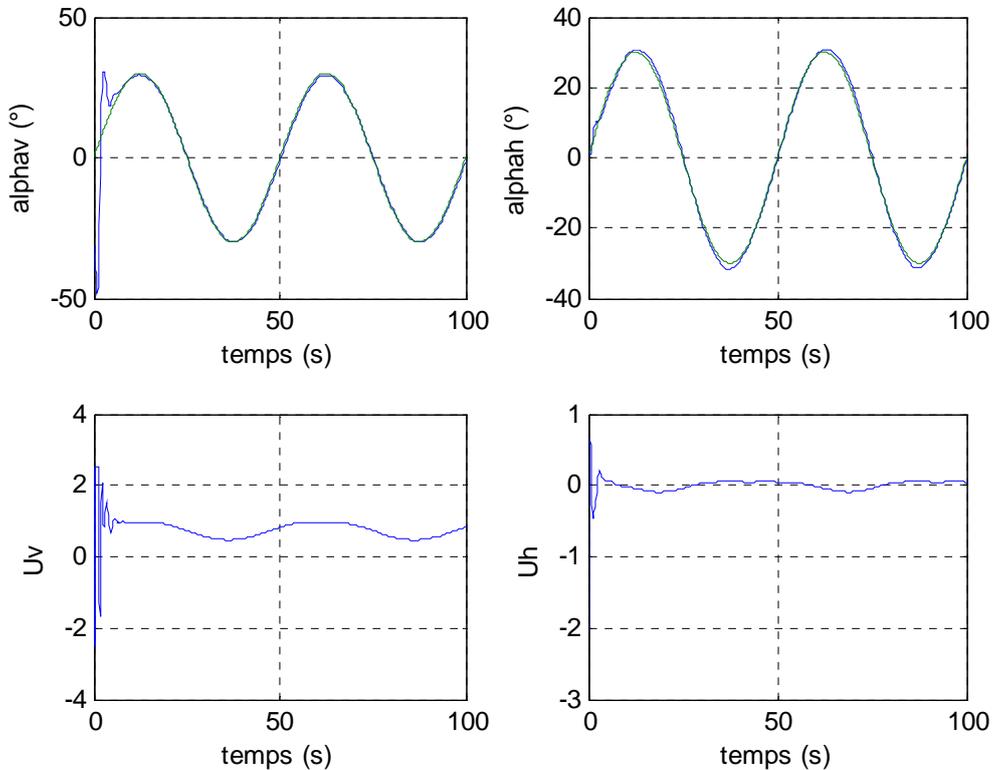


Figure V.27 : Fonctions d'activations floues des régulateurs par retour d'état pour un signal carré en référence.

### V.5.3 Commande multi contrôleurs $H^\infty$ avec commutateur flou

Les simulations suivantes montrent le suivi de trajectoire sinusoïdale du système avec une commande multi contrôleurs  $H^\infty$  couplée à un commutateur flou :



**Figure V.28** : réponses du système à une sinusoïde en référence avec une commande  $H^\infty$  et commutateur flou

Le commutateur flou a permis d'avoir des signaux de commande continus ne présentant aucun pic, de plus on constate une diminution des erreurs de poursuite.

La figure ci-dessous montre les fonctions d'activation des régulateurs, ce sont des fonctions continues, qui permettent un passage progressif d'un correcteur à un autre, plusieurs correcteurs sont donc actifs à la fois.

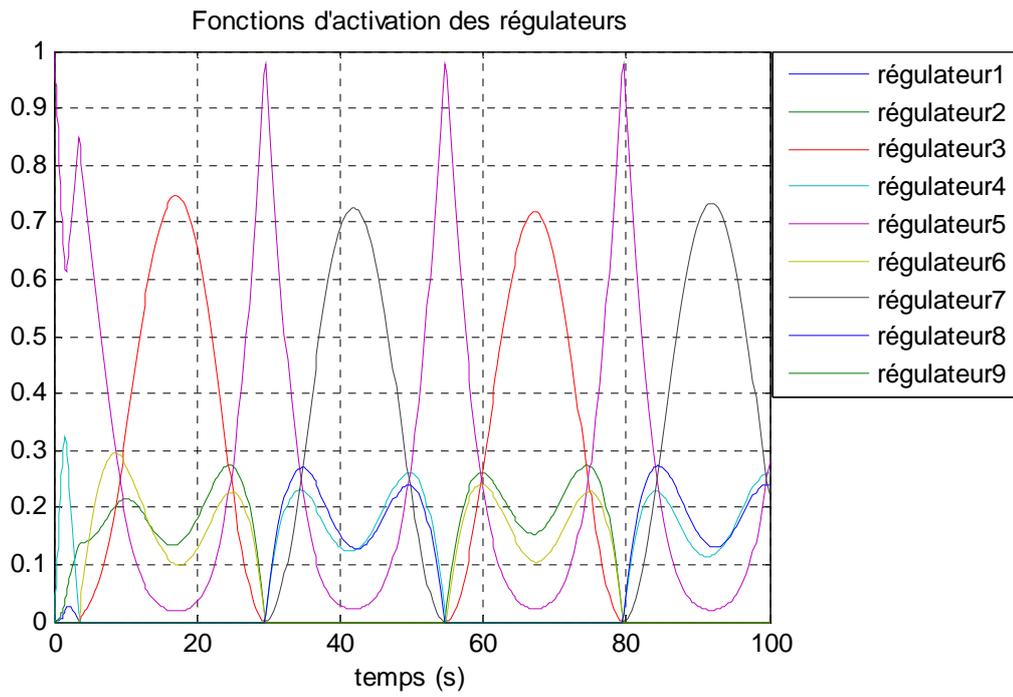


Figure V.29 : Fonctions d'activations floues des régulateurs  $H^\infty$

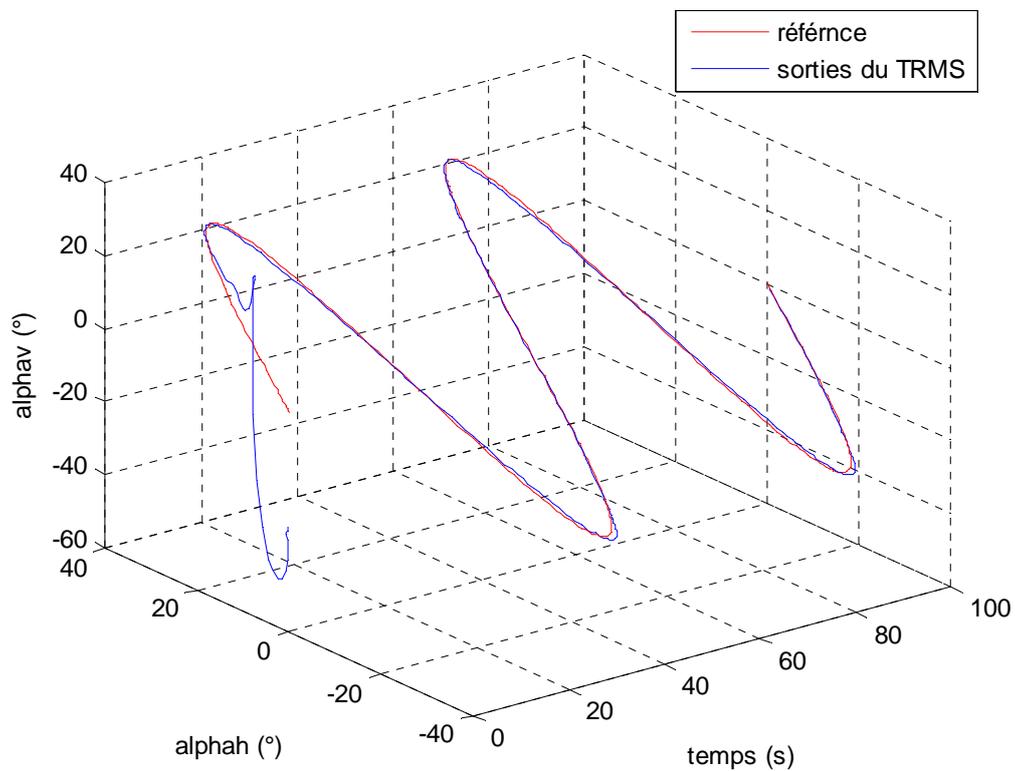


Figure V.30 : trajectoire du simulateur dans l'espace avec une commande  $H^\infty$  et commutateur flou

### V.5.4 Commande multi contrôleurs RST avec commutateur flou

Comme pour la commutation franche des contrôleurs RST, un commutateur flou est développé pour le sous-système d'élévation, celui-ci a également des fonctions d'activation triangulaire

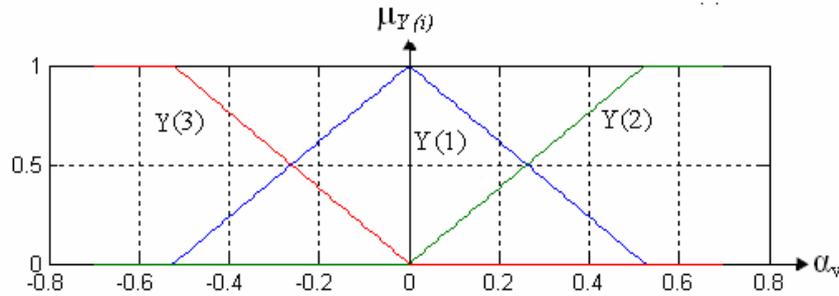


Figure V.31 : Partition floue relative à l'angle d'élévation

La base de règle est la suivante :

Si  $\alpha_v$  est  $Y(3)$  Alors :  $U=U_1$

Si  $\alpha_v$  est  $Y(1)$  Alors :  $U=U_2$

Si  $\alpha_v$  est  $Y(2)$  Alors :  $U=U_3$

La sortie floue est inférée de la manière suivante :

$$U_p = \frac{\sum_{i=1}^9 \mu_Y(\alpha_v) \times u_i}{\sum_{i=1}^9 \mu_Y(\alpha)} \quad (\text{V.11})$$

Les figures (V.32) (V.33) (V.34) et (V.35) montrent les sorties pour une référence en créneau et en sinusoïde ainsi que les fonctions d'activation des régulateurs :

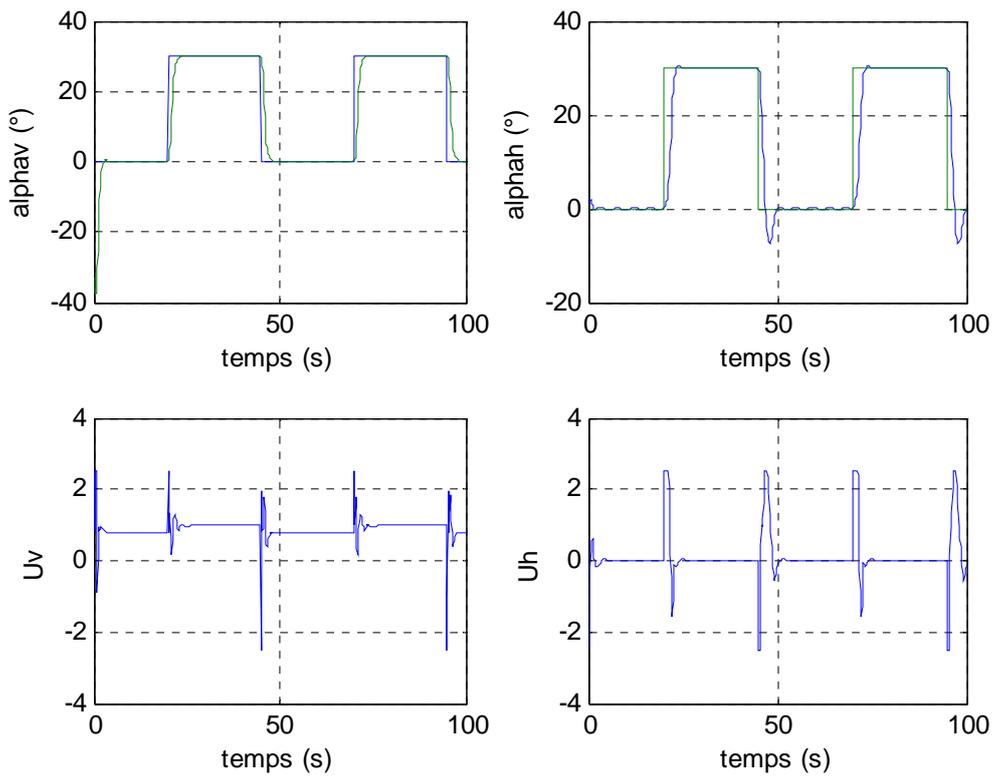


Figure V.32 : réponses du système et signaux de commande pour un signal de référence carré avec une commande multi contrôleur RST à commutation floue.

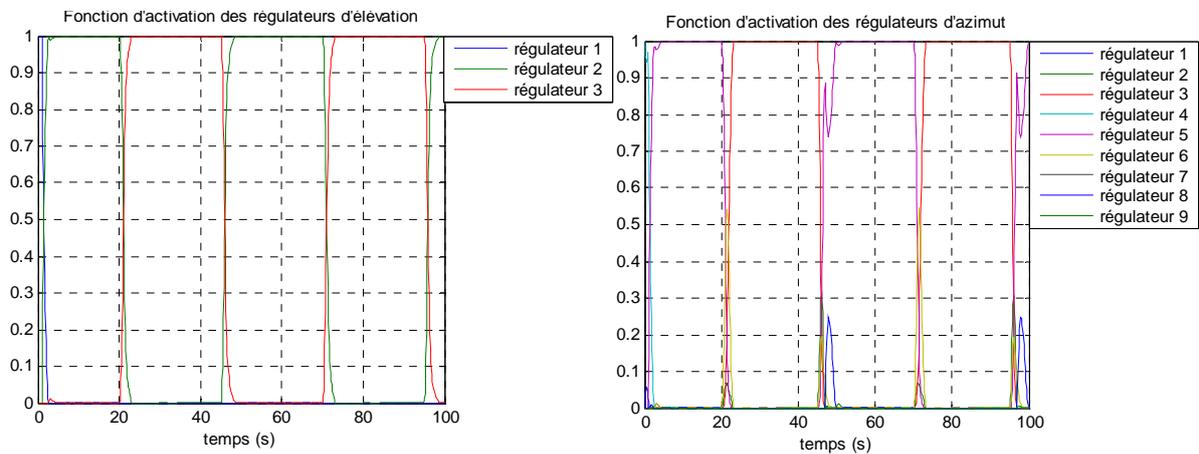


Figure V.33 : Fonctions d'activations floues des régulateurs RST pour un signal de référence carré.

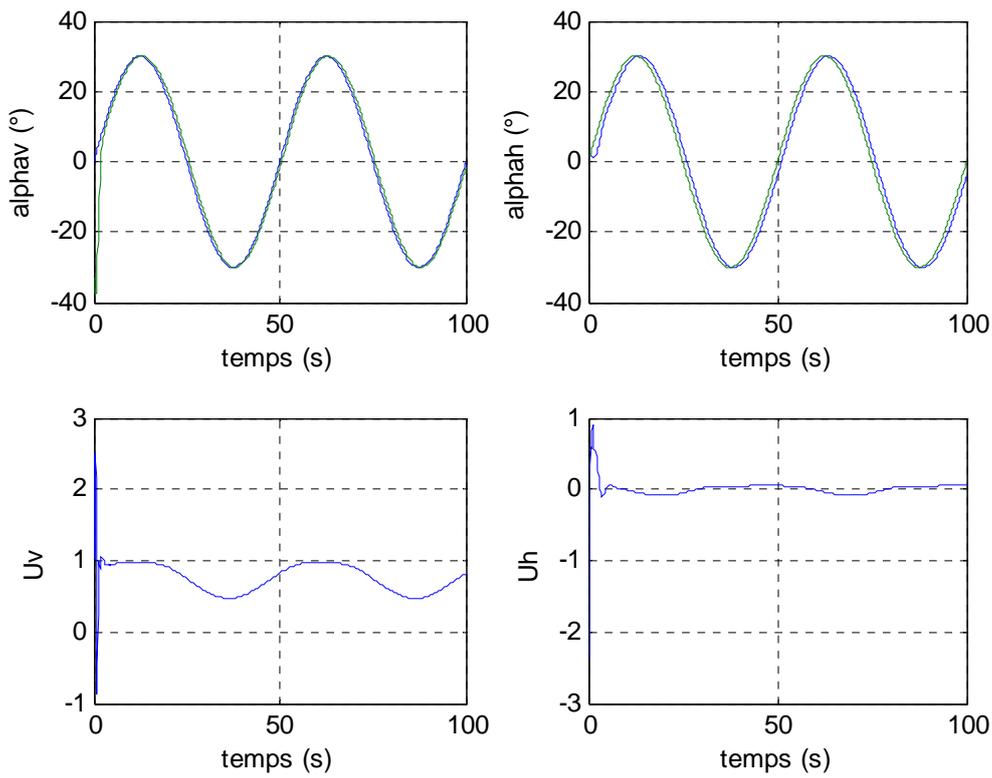


Figure V.34 : réponses du système et signaux de commande pour un signal de référence sinusoïdal avec une commande multi contrôleur RST à commutation floue

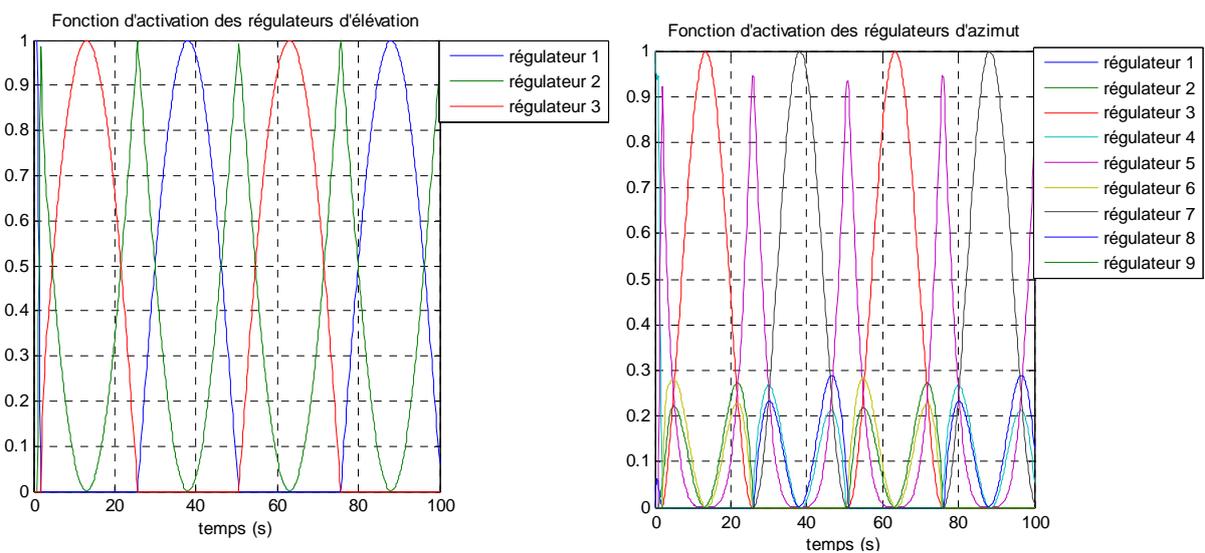
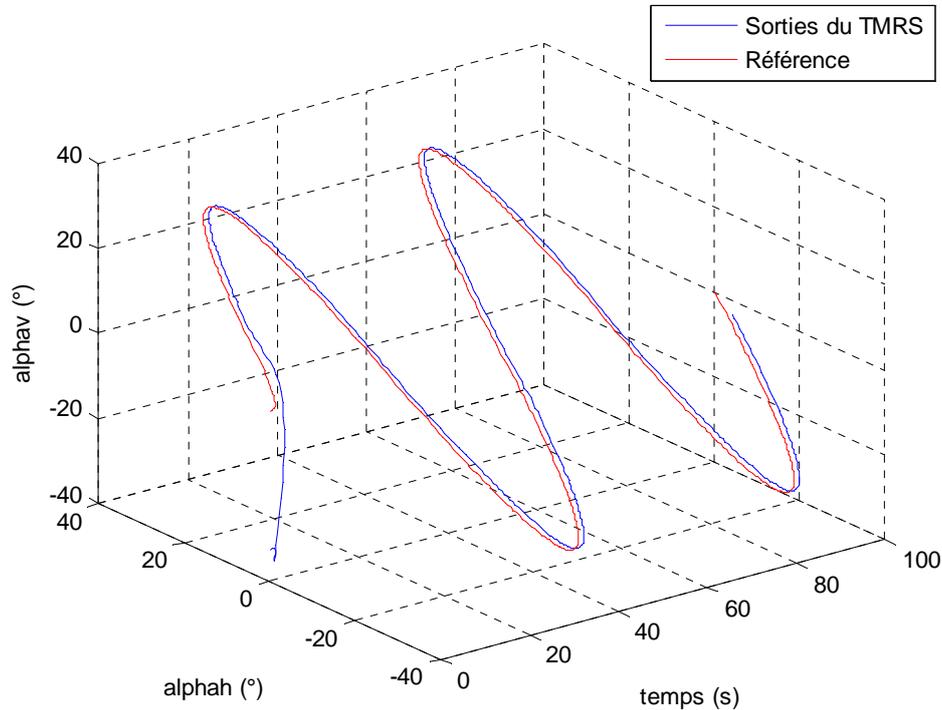


Figure V.35 : Fonctions d'activations floues des régulateurs RST pour un signal de référence sinusoïdal.



**Figure V.36 :** trajectoire dans l'espace d'une poursuite de sinusoides avec une commande multi contrôleurs RST à commutation floue.

En analysant les signaux de commande, on remarque la disparition des sauts et ce, grâce au passage flou d'un régulateur à l'autre. Ceci assure la continuité de la commande et supprime, ainsi, les bonds au niveau des sorties.

## V.6 Conclusion

Dans ce chapitre nous nous sommes intéressés, à l'application de la commande multi contrôleurs de type retour d'état,  $H_\infty$  et RST au simulateur d'hélicoptère. La commande multi contrôleurs par retour d'état était facile à synthétiser mais demande l'utilisation d'observateur d'état et n'a pas été assez performante pour suivre une sinusoïde de 0.02 Hertz. La commande multi contrôleurs du type  $H_\infty$  a présenté de bonnes performances de poursuite de sinusoïde et de signal carré, mais la difficulté de sa synthèse se situe au niveau du choix heuristique des matrices de pondération. Par contre, la commande multi contrôleur de type RST, qui a une synthèse assez simple, puisqu'elle se base sur les modèles à un degré de liberté, présente une

réponse rapide et sans dépassement. Cependant cette dernière a des performance moins bonnes (relativement à la commande multi contrôleur  $H^\infty$ ) concernant la poursuite de référence telle qu'une sinusoïde, où une erreur de suivie apparaît.

D'autre part, l'utilisation d'un commutateur à événement discret, bien qu'il applique au système le contrôleur adéquat, fait apparaître des discontinuités au niveau des tensions de commande. C'est le changement brusque de régulateur qui en est la cause. Ces sauts peuvent être supprimés par un commutateur flou dont le rôle consiste à passer d'une commande à une autre de façon douce.

## Chapitre VI

### *Implémentation et résultats expérimentaux*

#### VI.1 Introduction

Dans ce dernier chapitre sont exposés les résultats expérimentaux obtenus après l'implémentation en temps réel des commandes synthétisées précédemment.

Ces commandes ont été testées sur le simulateur d'hélicoptère pour satisfaire les objectifs de stabilisation et de poursuite de trajectoire sinusoïdale. Et afin de tester la robustesse des commandes appliquées au système, des variations paramétriques ont été introduites. Celles-ci consistent à faire pivoter le contre poids le long de la tige qui le supporte. Ceci a pour effet le déplacement du centre de gravité du procédé.

Par ailleurs, en impulsant la position de la poutre le système subi une perturbation (au niveau du contre poids). Cette expérience est effectuée dans le but de tester ses performances vis-à-vis du rejet de perturbations.

#### VI.2 Considérations pratiques

Comme notre système est commandé via un calculateur numérique (PC), le choix de la fréquence d'échantillonnage s'impose. Pour ce faire, cette dernière doit être choisie assez grande de manière à approcher le comportement continu des régulateurs synthétisés précédemment. De plus, un compromis entre le temps de calcul et la période d'échantillonnage doit être déterminé comme le montre la figure suivante :

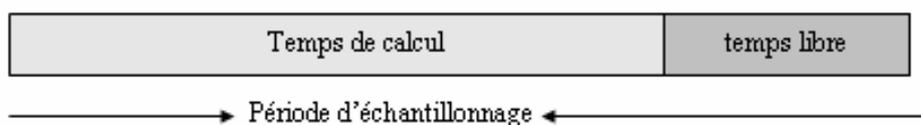


Figure VI.1 : temps de calcul et période d'échantillonnage

Ainsi on doit avoir :

$$T_{\text{calcul}} < T_{\text{échantillonnage}}$$

et  $T_{\text{échantillonnage}}$  suffisamment petite  $\Rightarrow$  travail en pseudo continu

La période d'échantillonnage retenue pour l'implémentation est :  $T_{\text{échantillonnage}}=0.001$  seconde.

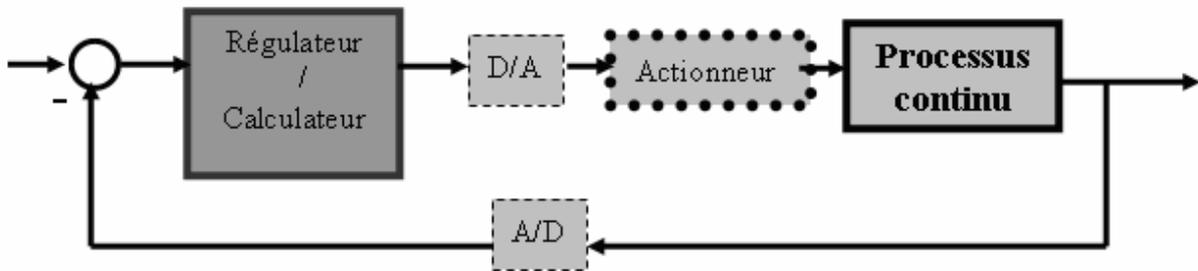


Figure VI.2 : boucle de régulation échantillonnée

Les capteurs utilisés pour la mesure des angles sont des capteurs optiques incrémentaux. Ainsi, et avant l'exécution du programme de commande, il faut veiller à initialiser les angles d'élévation et d'azimut à la position (0,0), comme le montre la figure ci-dessous :

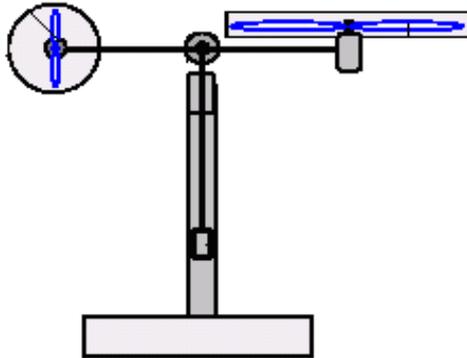


Figure VI.3 : initialisation dans la position (0,0)

### VI.3 Implémentation des commandes avec commutateur à événements discrets

#### VI.3.1 Commande multi contrôleurs $H_\infty$ avec commutation franche

Les figures suivantes montrent les réponses expérimentales obtenues en utilisant la commande multi contrôleurs  $H_\infty$ .

##### VI.3.1.1 Stabilisation et poursuite

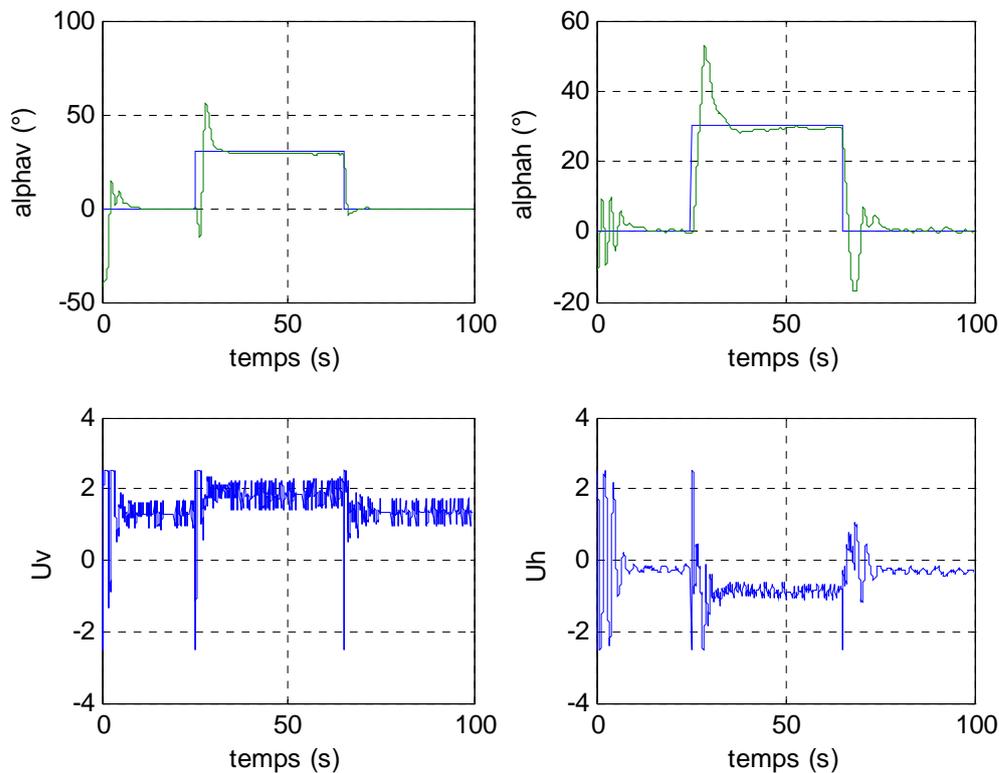


Figure VI.4 : stabilisation sur une référence carrée avec commande  $H_\infty$  CED

On constate que la commande a pu stabiliser le simulateur suivant des références en créneaux avec des temps de réponse avoisinant les 6 secondes. Par ailleurs, on observe des dépassements de l'ordre de 20 %.

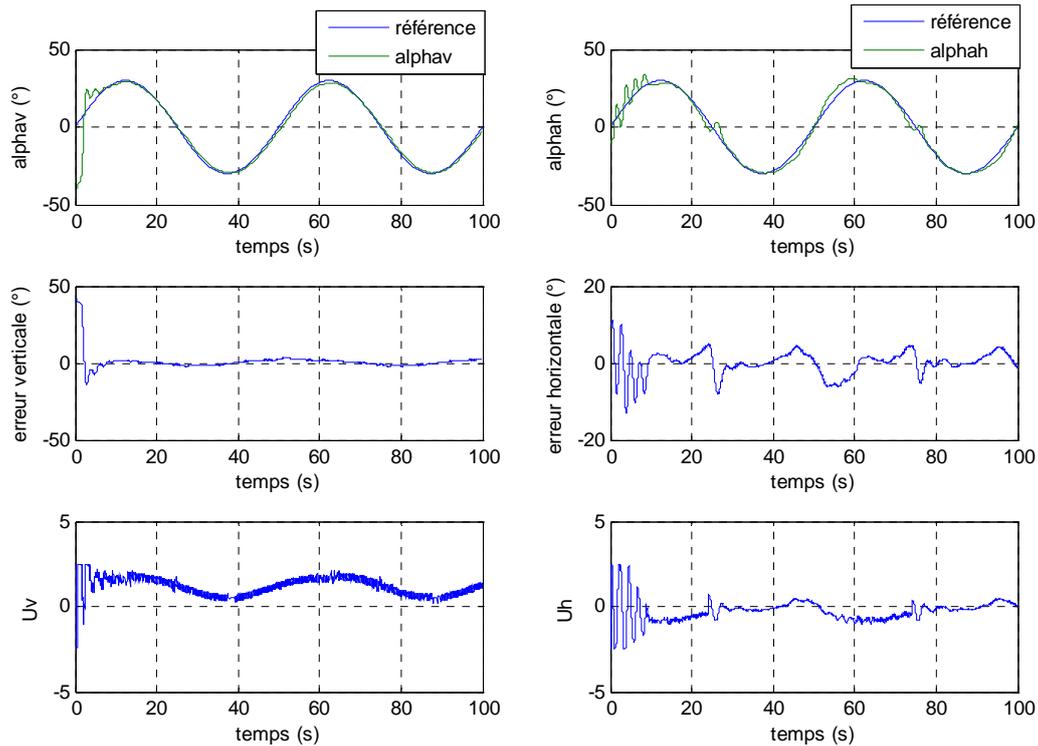


Figure VI.5 : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande  $H_{\infty}$  CED

La commande a pu assurer la poursuite de trajectoire sinusoïdale de fréquence 0.02Hz avec des erreurs de poursuites de l'ordre de  $3^{\circ}$  pour le vertical et de  $5^{\circ}$  pour l'horizontal. De plus, l'angle d'azimut présente des oscillations au début de l'expérience du fait de la faible inertie du mouvement horizontal, contrairement au mouvement vertical où la gravité réduit les dépassements de l'angle d'élévation. Ce phénomène rend l'asservissement de l'angle d'azimut difficile conduisant ainsi à l'augmentation de l'erreur de poursuite.

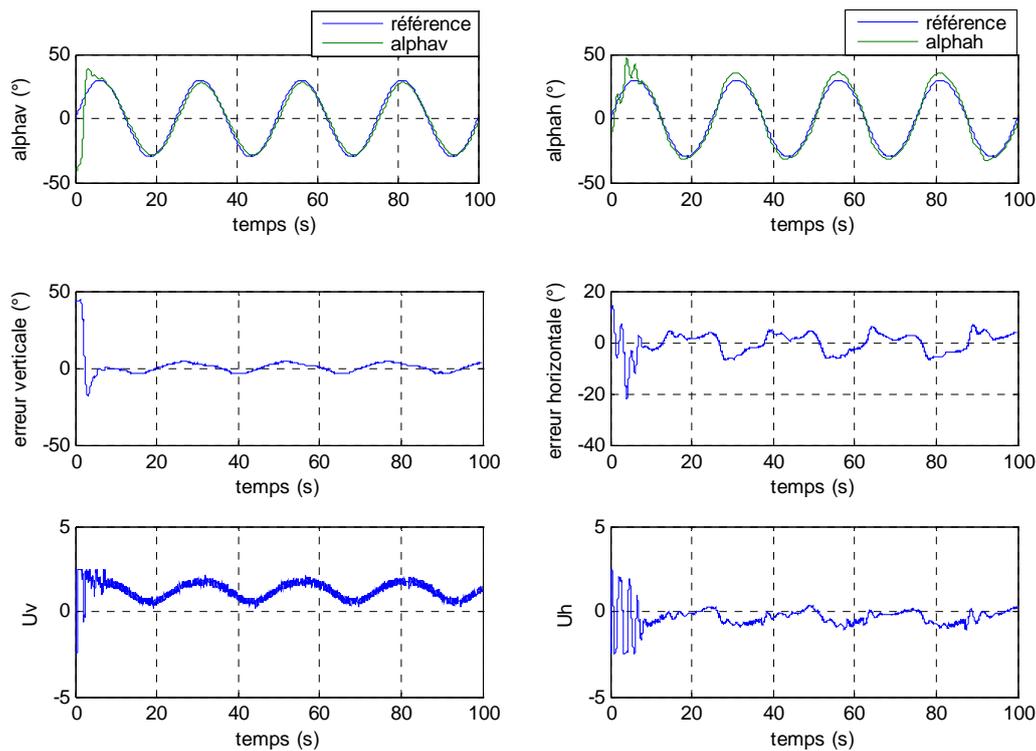


Figure VI.6 : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande  $H_{\infty}$  CED

La poursuite d'une sinusoïde plus rapide (0.04Hz) est assurée. Cependant, des erreurs de poursuite un peu plus grandes apparaissent.

Les signaux de commande restent dans les limites tolérées par les actionneurs, néanmoins ils présentent des pics dus à la commutation franche. Ces sauts sont filtrés par le système (leur effet n'apparaît que très peu à la sortie). Cependant, on peut améliorer les réponses par une commutation floue qui sera exposée ultérieurement.

### VI.3.1.2 Variations paramétriques

Outre la variation du centre de gravité de la structure par le déplacement du contre poids, il existe d'autres variations qu'on ne peut contrôler. Ces dernières n'ont pas été prises en considération lors de l'établissement du modèle mathématique. Parmi ces perturbations paramétriques figure la force de torsion des câbles d'alimentation des moteurs et des capteurs. De plus, la structure est composée de parties qui ne peuvent être fixées rigidement, comme les boucliers de protection des hélices, qui vibrent lors de la rotation de ces dernières, sans compter la position dans laquelle est placé le simulateur par rapport aux murs, qui renvoient l'air généré par les propulseurs. Ce phénomène touche particulièrement le rotor de queue, qui

subit une rafale transversale qui heurte le simulateur l'obligeant, ainsi, à corriger fréquemment sa trajectoire.

Cependant, les réponses ci-dessous montrent que la commande a pu faire face à tous ces phénomènes qui rendent notre système incertain.

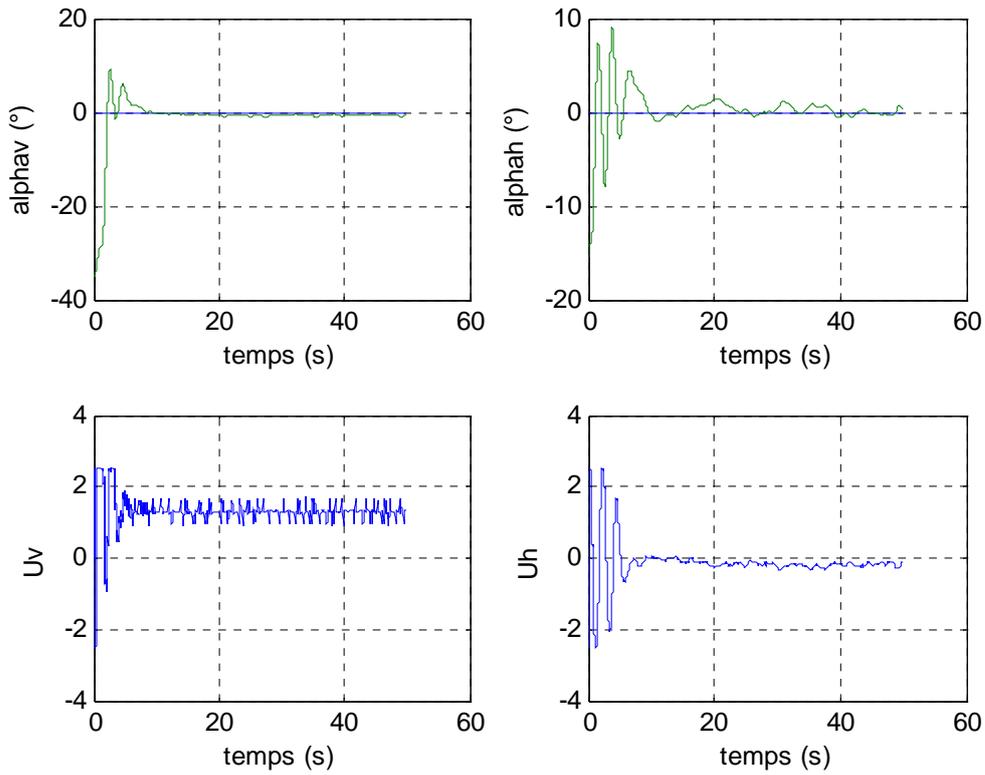


Figure VI.7 : stabilisation en présence de variations paramétriques avec commande  $H_\infty$  CED

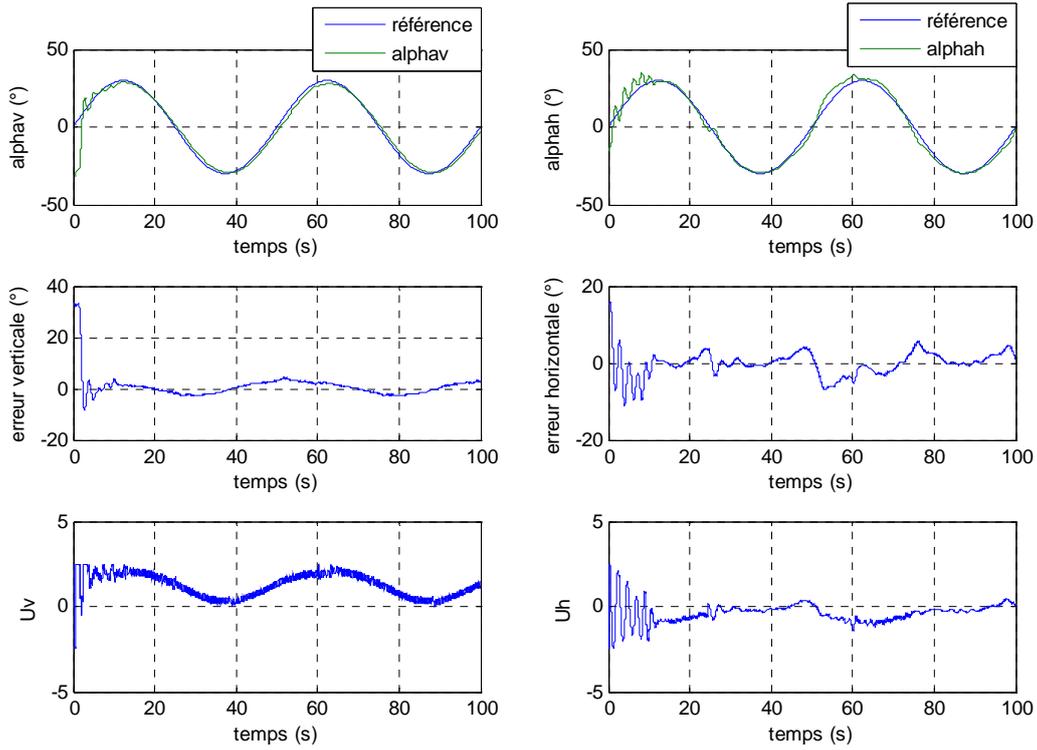


Figure VI.8 : poursuite d'une sinusoïde de 0.02 Hz en présence de variations paramétriques avec commande  $H_\infty$  CED

VI.3.1.3 Rejet de perturbations

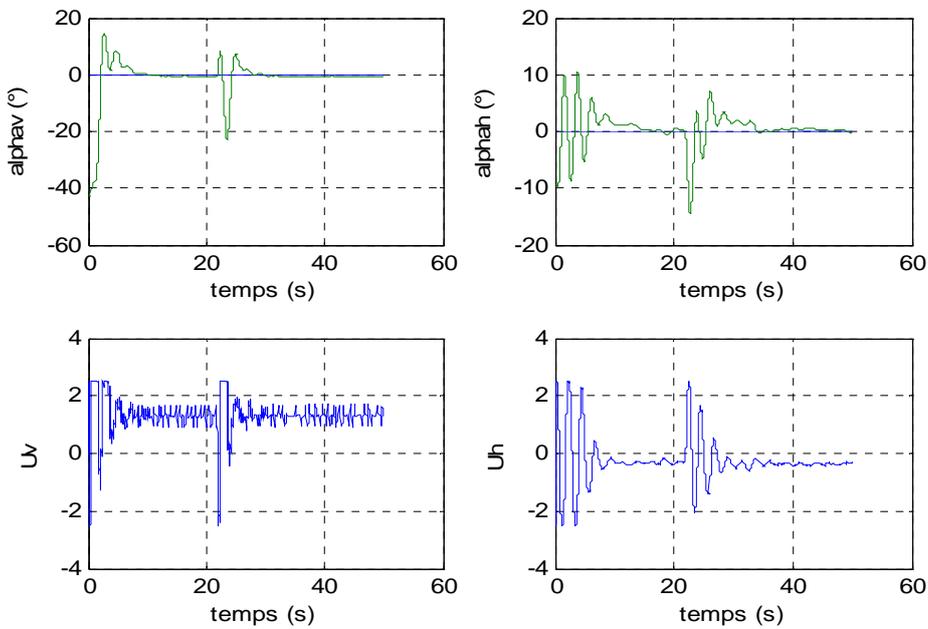


Figure VI.9 : rejet de perturbations avec la commande multi contrôleurs  $H_\infty$  CED

La commande multi contrôleurs  $H_\infty$  à commutation franche a pu rejeter une perturbation introduite au système après un temps relativement court.

### VI.3.2 Commande multi contrôleurs RST avec commutation franche

La commande appliquée par la suite est une commande à trois branches (commande RST), cette dernière synthétisée sur la base d'un modèle de référence, nous a permis d'avoir des stabilisations avec des dépassements plus réduits que ceux relatifs à la commande  $H_\infty$ .

#### VI.3.2.1 Stabilisation et poursuite

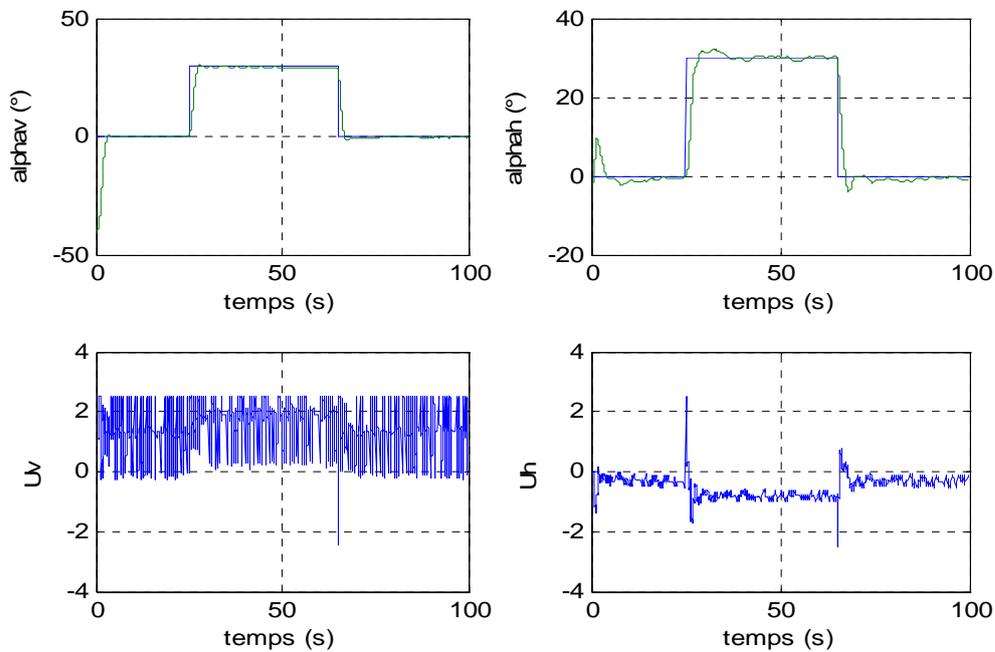


Figure VI.10 : stabilisation sur une référence carrée par une commande RST CED

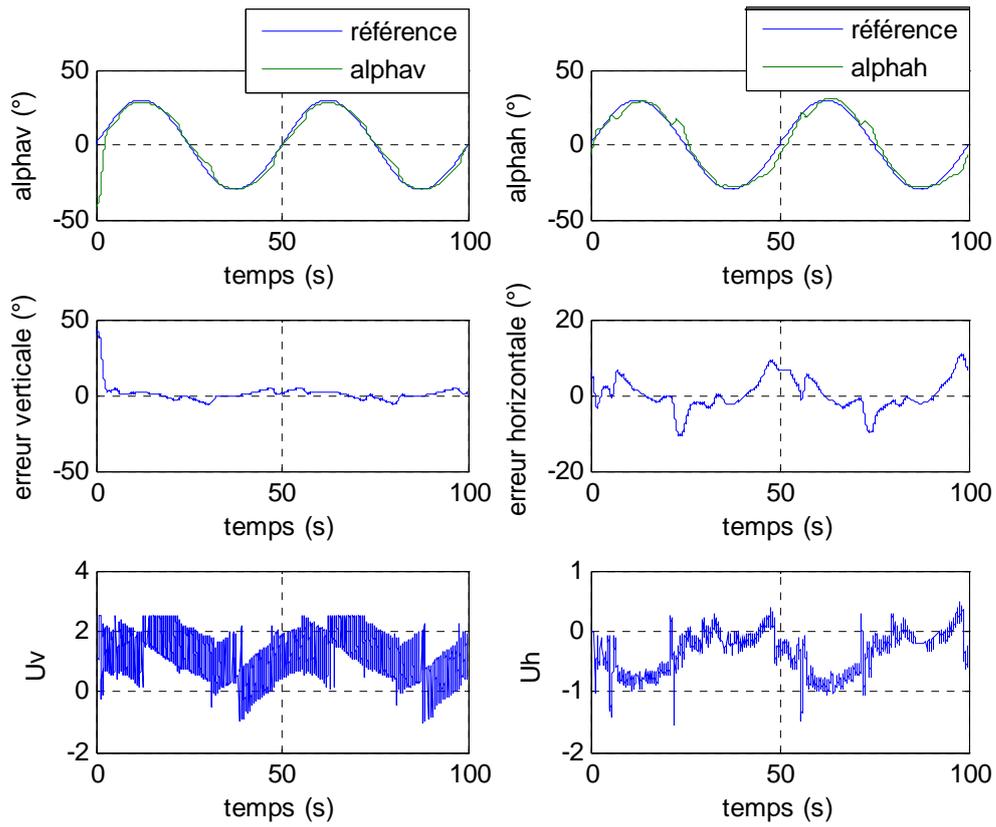


Figure VI.11 : poursuite d'une sinusoïde de 0.02 Hz par une commande RST CED

La poursuite de sinusoïde présente des sauts qui apparaissent, nettement plus, que pour la commande  $H_\infty$ . Ces bonds qui sont dus aux commutations franches d'un régulateur à un autre, vont être éliminés par la suite avec l'utilisation de la commutation floue.

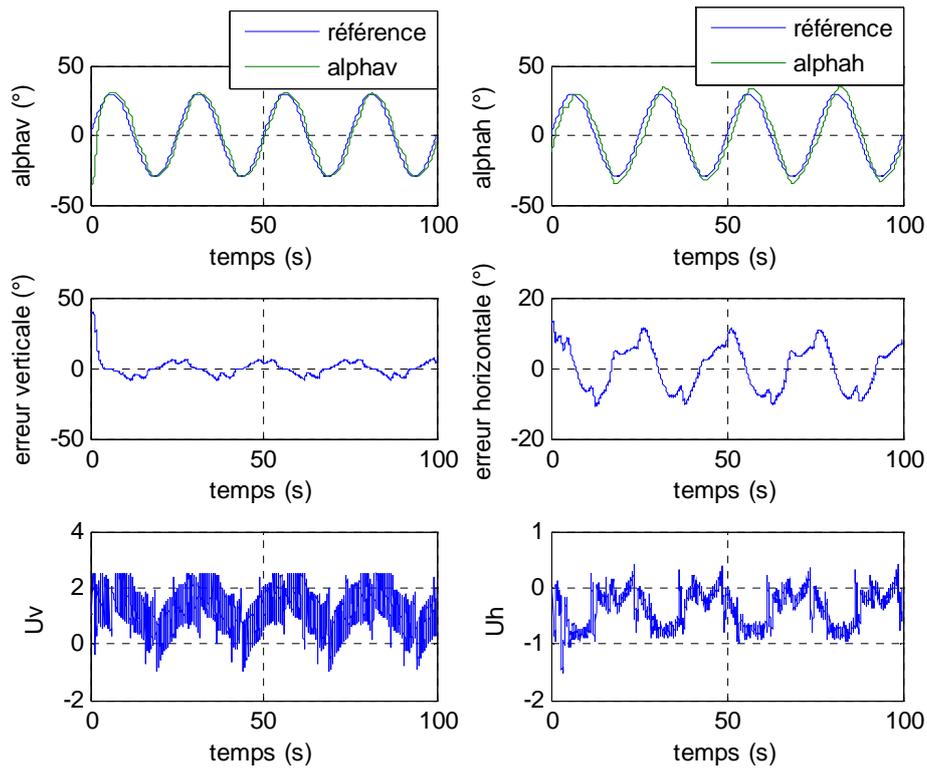


Figure VI.12 : poursuite d'une sinusoïde de 0.04 Hz par une commande RST CED

On remarque une augmentation des erreurs de suivi lors de la poursuite d'une sinusoïde à fréquence plus élevée.

VI.3.2.2 Variation paramétriques

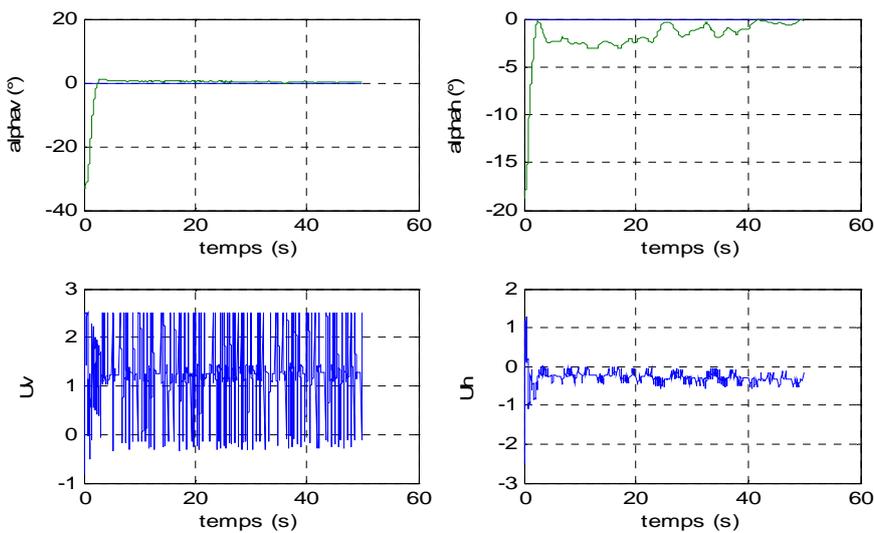


Figure VI.13 : stabilisation par une commande RST CED en présence de variations paramétriques

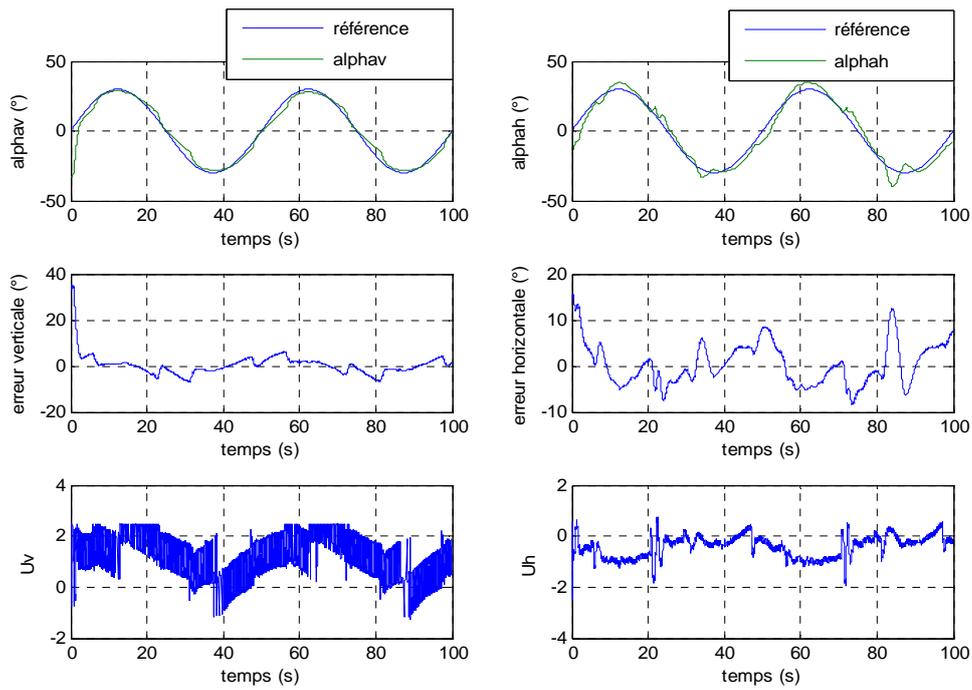


Figure VI.14 : poursuite d'une sinusoïde 0.02 Hz par une commande RST CED en présence de variations paramétriques

On remarque que le déplacement du contre poids a perturbé le mouvement horizontal, introduisant ainsi une erreur entre le point de référence 0° et l'angle d'azimut.

VI.3.2.3 Rejet de perturbations

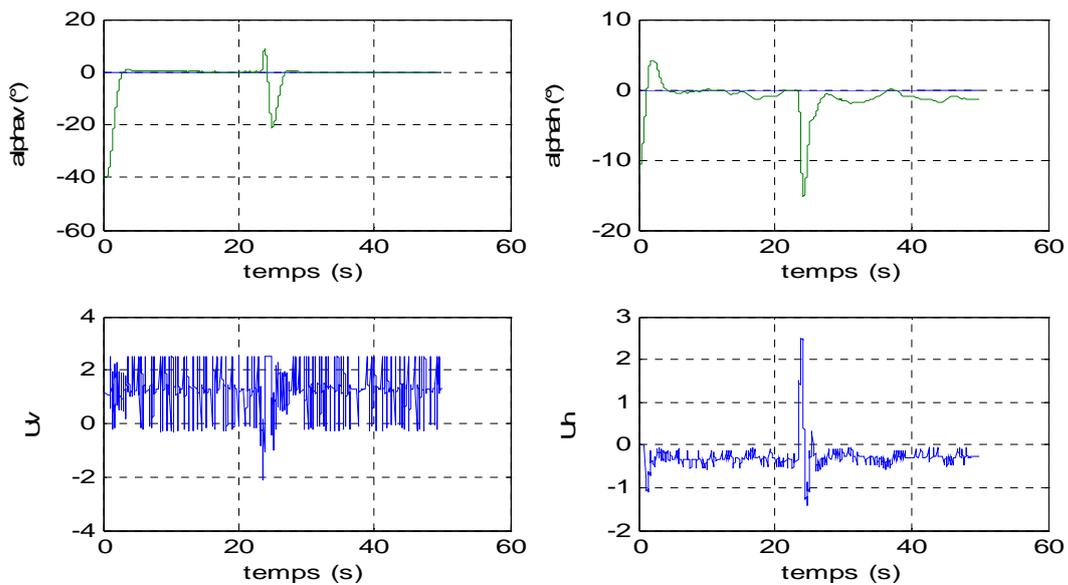


Figure VI.15 : rejet de perturbations par une commande RST CED

Le rejet de perturbation était assez rapide, sans apparition d'erreur statique sur l'angle vertical. L'angle horizontal présente néanmoins quelques erreurs permanentes.

## VI.4 Implémentation des commandes avec commutateur flou

Afin d'éliminer les discontinuités dues aux commutations brusque qui introduisent des sauts dans les réponses du système, cette partie consiste à implémenter les commandes multi contrôleurs précédentes avec un commutateur qui utilise la logique floue pour la sélection des contrôleurs adéquats.

### VI.4.1 Commande multi contrôleurs $H_\infty$ avec commutation floue

Les figures ci-dessous montrent les réponses obtenues avec la commande  $H_\infty$  couplée à un commutateur flou.

#### VI.4.1.1 Stabilisation et poursuite

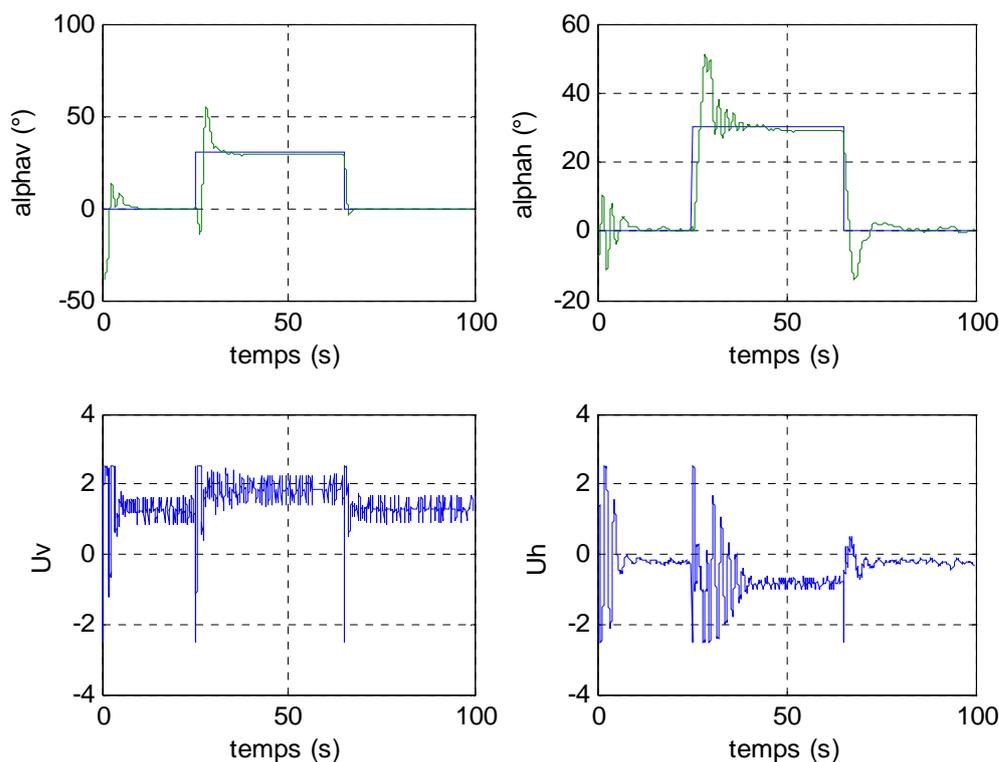


Figure VI.16 : stabilisation sur une référence carrée avec commande  $H_\infty$  floue

On observe que la stabilisation reste pratiquement inchangée.

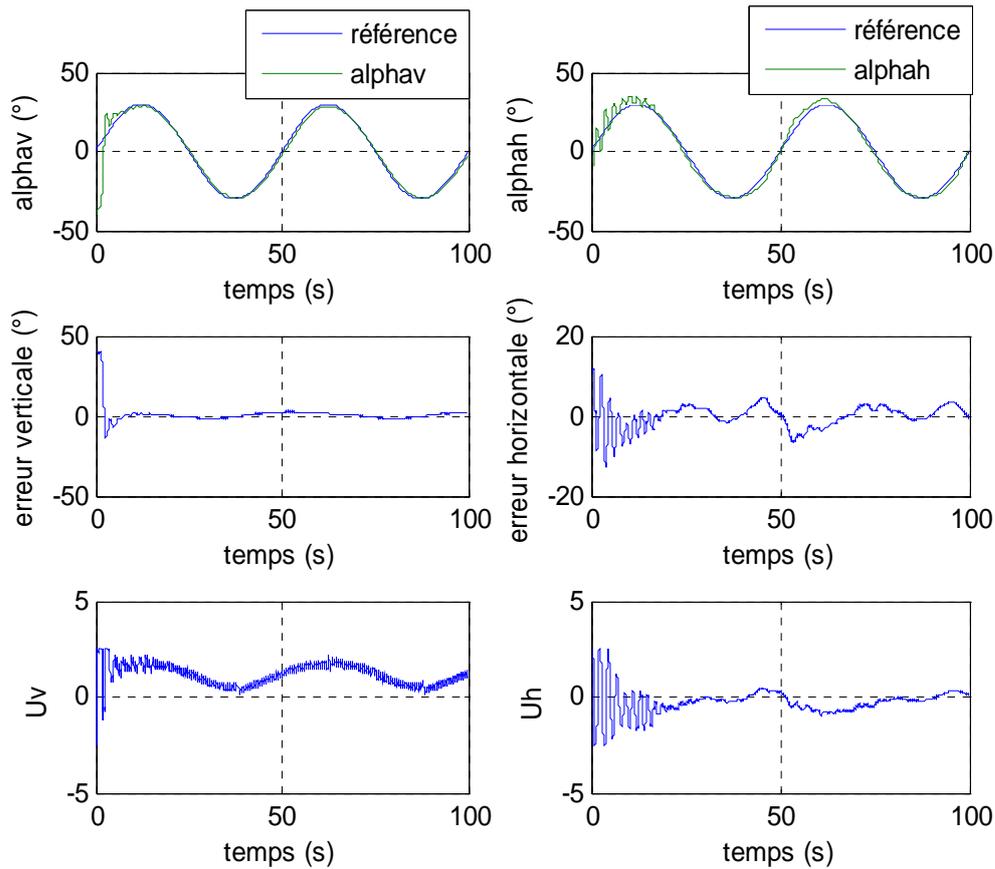


Figure VI.17 : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande  $H_{\infty}$  floue

Par contre, les commandes relatives à la poursuite de sinusoïde sont douces et continues, ce qui a pour effet l'élimination des sauts dans les réponses et une diminution des erreurs de poursuite. Néanmoins, la poursuite de trajectoire sinusoïdale présente des oscillations dans le régime transitoire (début du mouvement) dues essentiellement à des phénomènes non modélisés et au couplage. Leur effet se fait sentir sur les deux sorties du système, plus particulièrement sur le mouvement horizontal.

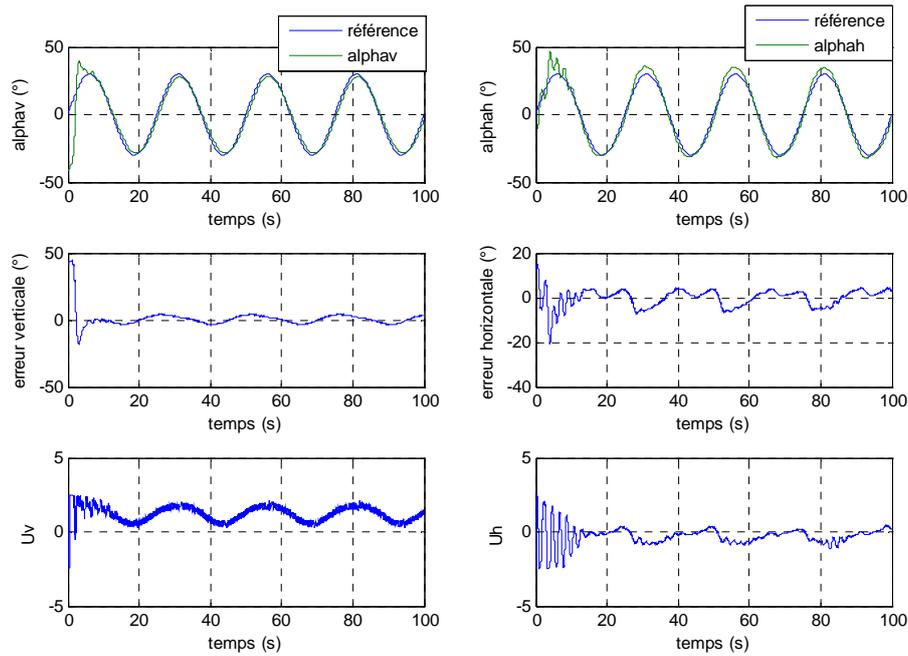


Figure VI.18 : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande  $H_\infty$  floue

La poursuite d'une sinusoïde de fréquence plus importante (0.04 Hz) fait apparaître des erreurs de poursuite un peu plus importantes, mais reste néanmoins bonne.

VI.4.1.2 Variations paramétriques

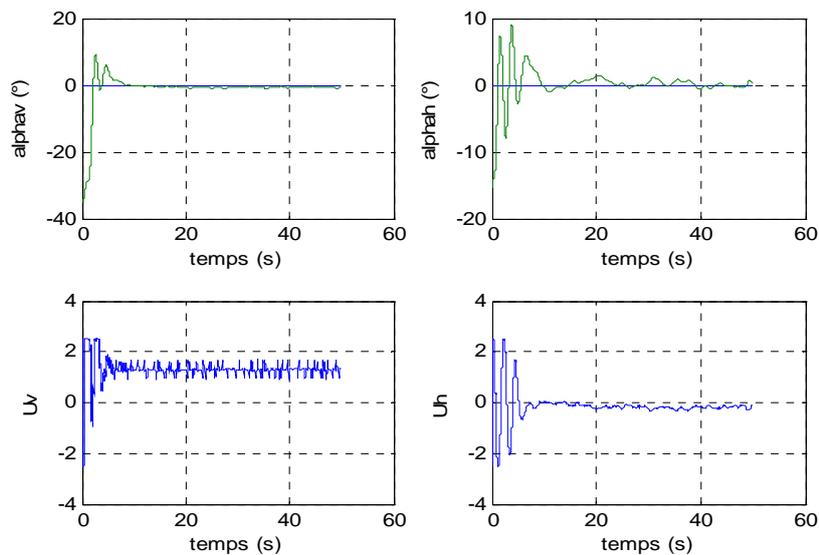


Figure VI.19 : stabilisation en présence de variations paramétriques avec commande  $H_\infty$  floue

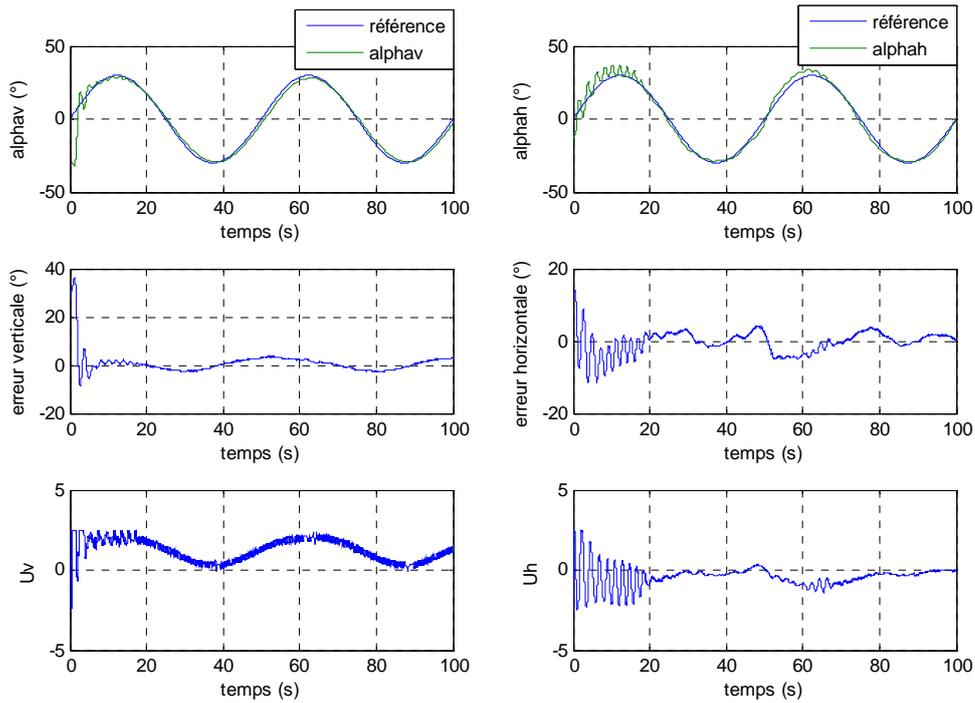


Figure VI.20 : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande  $H_\infty$  floue en présence de variations paramétriques

VI.4.1.3 Rejet de perturbations

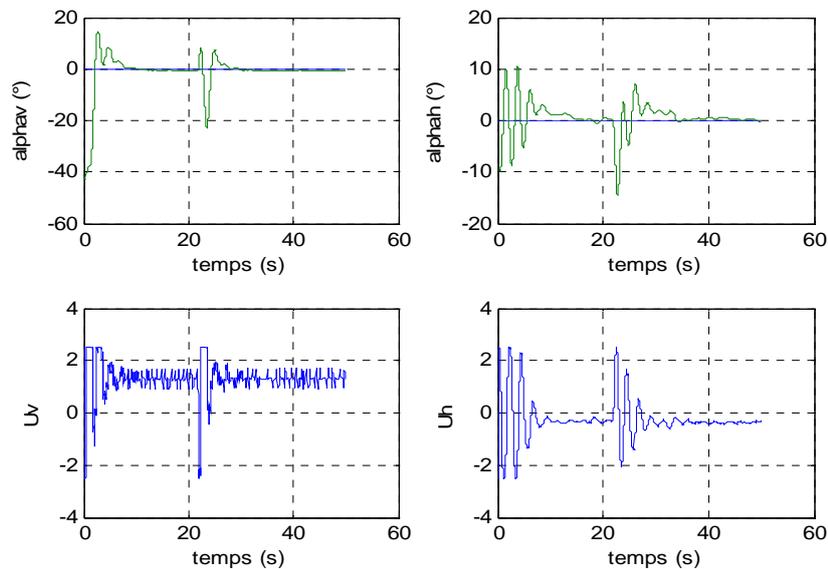


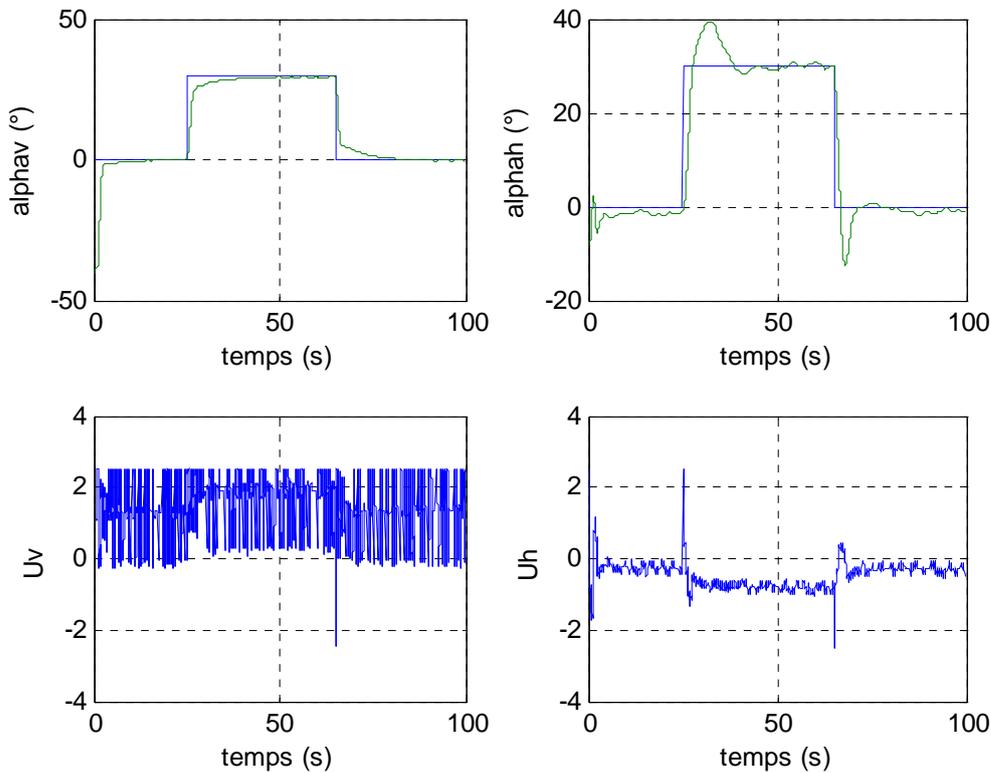
Figure VI.21 : rejet de perturbations par une commande  $H_\infty$  floue

Les performances et la robustesse restent sensiblement inchangées par rapport à la commande multi contrôleurs  $H_\infty$  avec commutateur à événements discrets (CED). Par contre, pour la poursuite de trajectoire sinusoïdale, les résultats sont meilleurs que ceux obtenus pour la commutation franche.

**VI.4.2 Commande multi contrôleurs RST avec commutation floue**

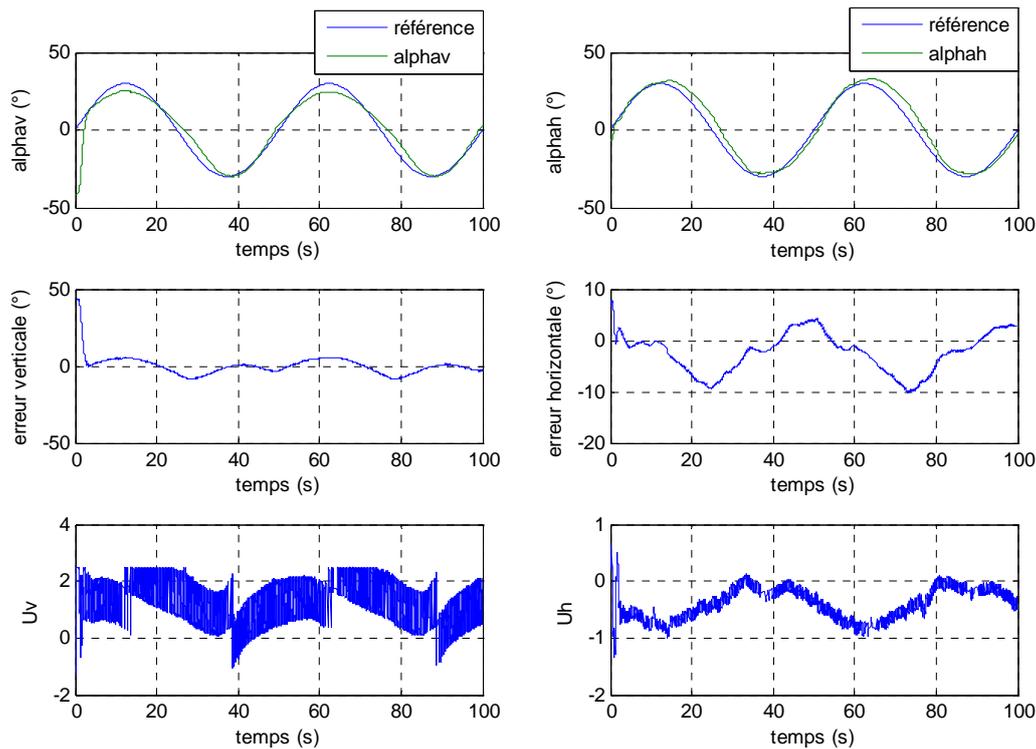
De la même façon, on applique la commutation floue à la commande multi contrôleurs RST, et les résultats expérimentaux sont exposés ci-dessous.

**VI.4.2.1 Stabilisation et poursuite**



**Figure VI.22 :** stabilisation sur une référence carrée avec commande RST floue

On remarque une amélioration de la stabilisation de l'angle d'élévation, par contre, on remarque l'apparition de dépassements sur l'angle azimut.



**Figure VI.23 :** poursuite d'une sinusoïde de 0.02 Hz par une commande RST floue

La combinaison de la commutation floue avec la commande multi contrôleurs *RST* a permis d'avoir une amélioration de la poursuite de l'angle d'azimut où il n'y a pratiquement pas d'oscillations au début du mouvement (constatés pour les autres commandes). Par contre, une erreur de poursuite apparaît dans le mouvement vertical (l'angle d'élévation).

Ainsi, la caractéristique de la commande *RST* qu'on appelle aussi commande à trois branches grâce à ses deux filtres : filtrage de la consigne, filtrage de la sortie, (voir chapitre *IV*), couplée à un commutateur flou nous a permis de mieux commander le rotor de queue (angle d'azimut). Celui-ci présentait des oscillations et des dépassements lors de l'utilisation des autres commandes.

L'erreur de poursuite augmente avec l'augmentation de la fréquence de la sinusoïde, comme le montre la figure VI.23. Cela est dû au couplage important entre les deux mouvements.

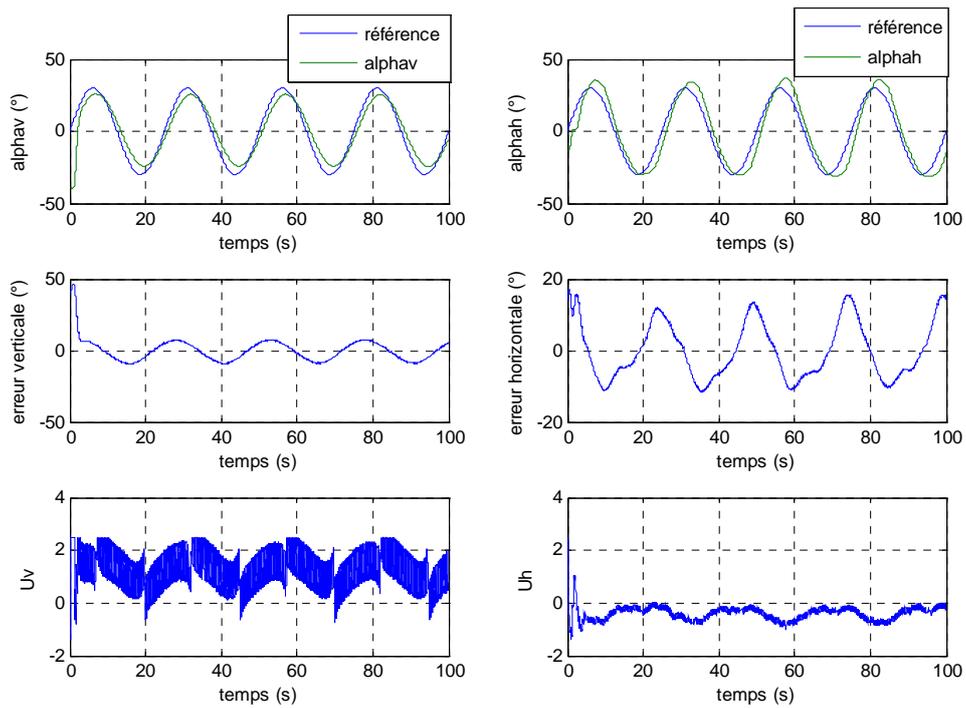


Figure VI.23 : poursuite d'une sinusoïde de 0.04 Hz par une commande RST floue

VI.4.2.2 Variations paramétriques

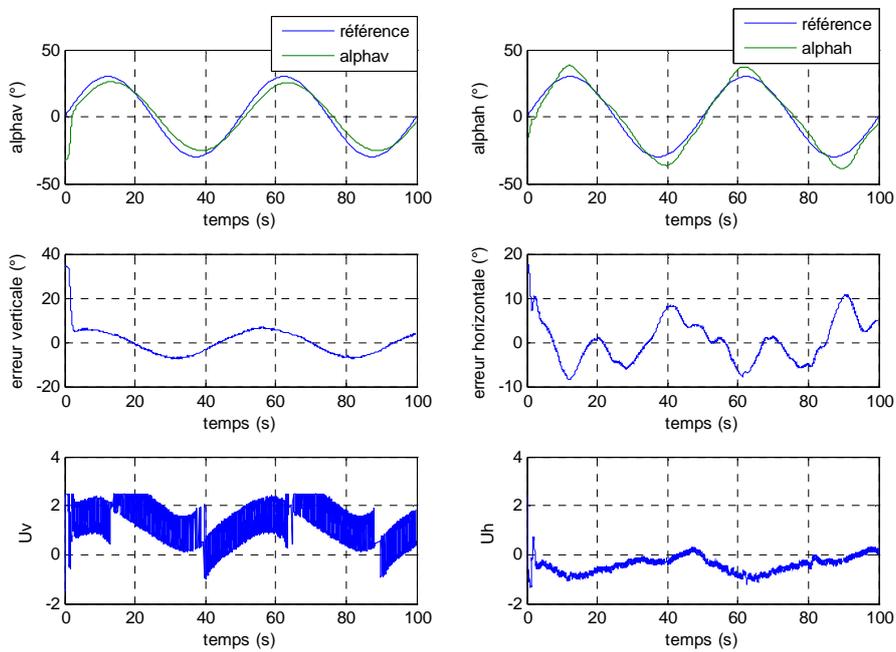


Figure VI.24 : poursuite d'une sinusoïde de 0.02 Hz par une commande RST floue en présence de variations paramétriques

## VI.4.2.3 Rejet de perturbations

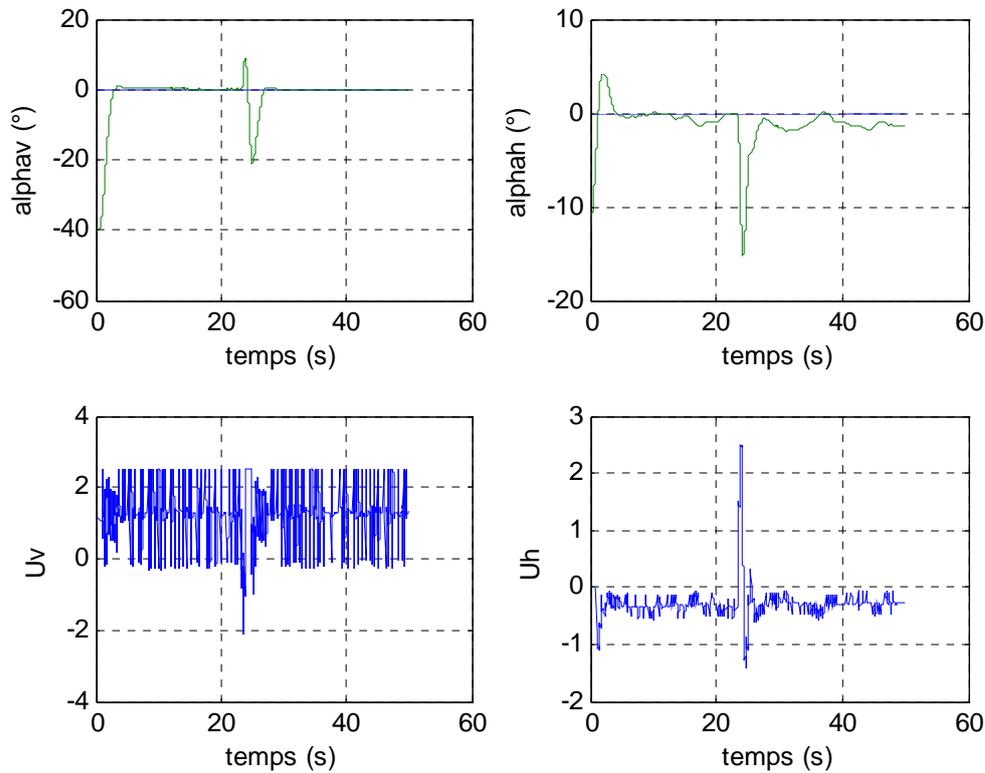


Figure VI.25 : rejet de perturbations par une commande RST floue

## VI.5. Superposition des réponses réelle et des réponses théoriques

Afin de pouvoir valider les résultats obtenus en théorie à travers les simulations obtenues avec le modèle mathématique du simulateur d'hélicoptère, les figures suivantes présentent le tracé des réponses réelles et des réponses théoriques du modèle TRMS.

On remarque une superposition parfaite entre les réponses théoriques et les réponses pratiques, en régime établi (régime permanent  $\omega=0$ ). Cependant, il y a des différences en régime transitoire (hautes fréquences), dues à la bonne connaissance du système en basse fréquence où le modèle mathématique décrit bien le comportement du procédé. Toutefois ; il subsiste aux hautes fréquences quelques phénomènes non modélisés, ce qui rend le système méconnu dans cette bande de fréquences.

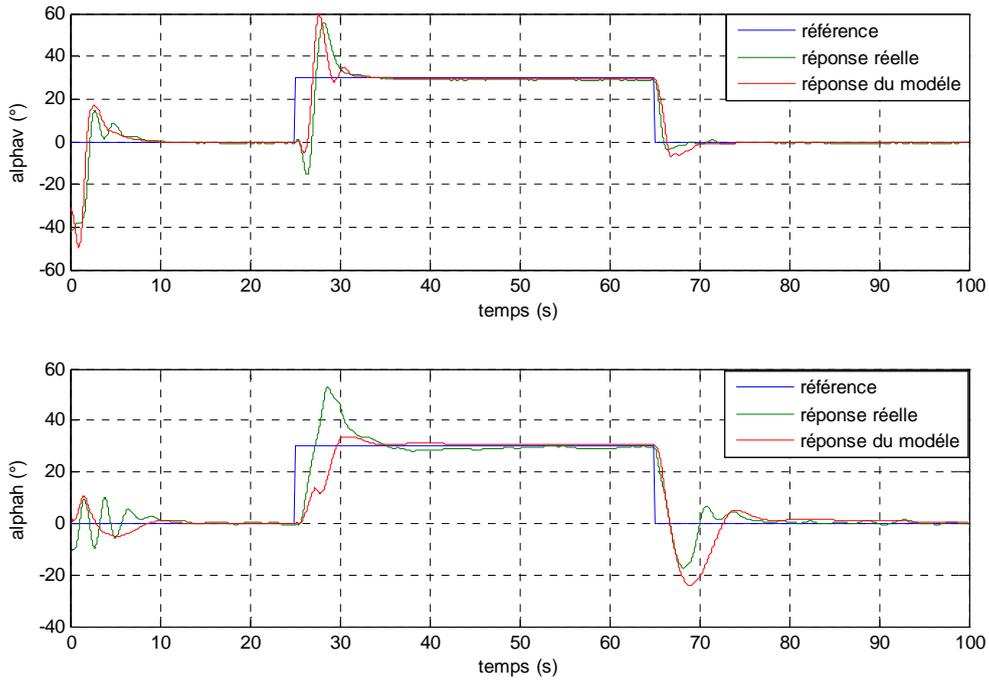


Figure VI.26 : réponse réelle avec réponse du modèle (commande  $H_{\infty} CED$ )

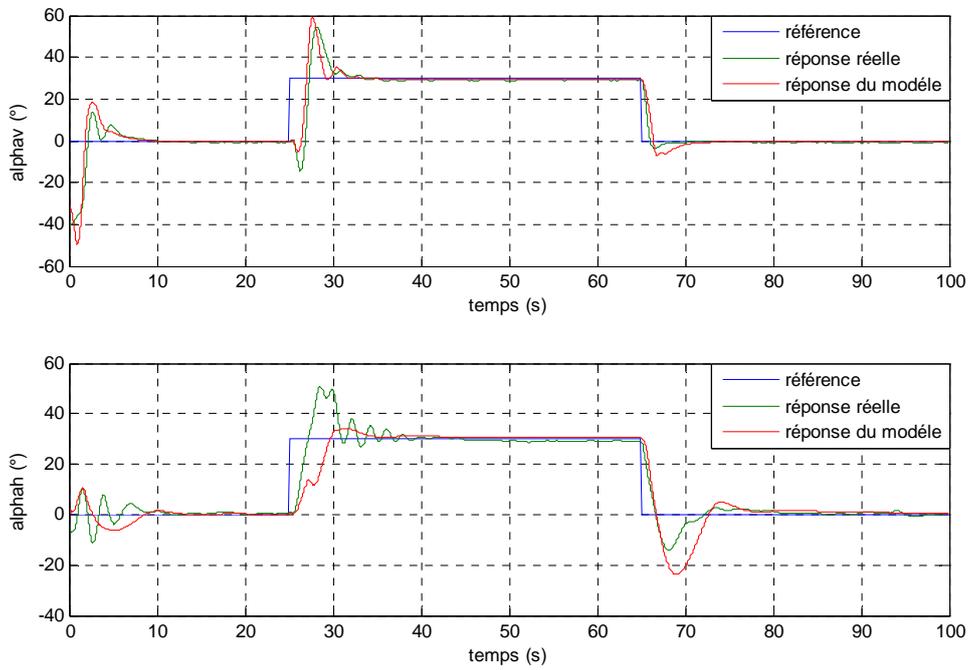


Figure VI.27 : réponse réelle avec réponse du modèle (commande  $H_{\infty} flow$ )

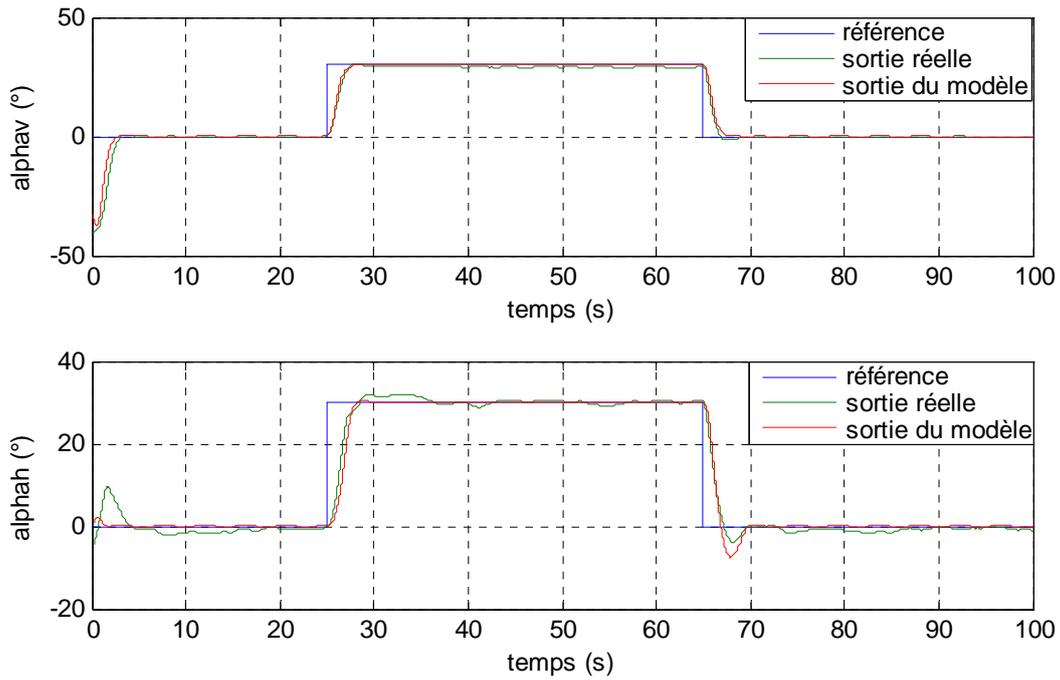


Figure VI.28 : réponse réelle avec réponse du modèle (commande RST CED)

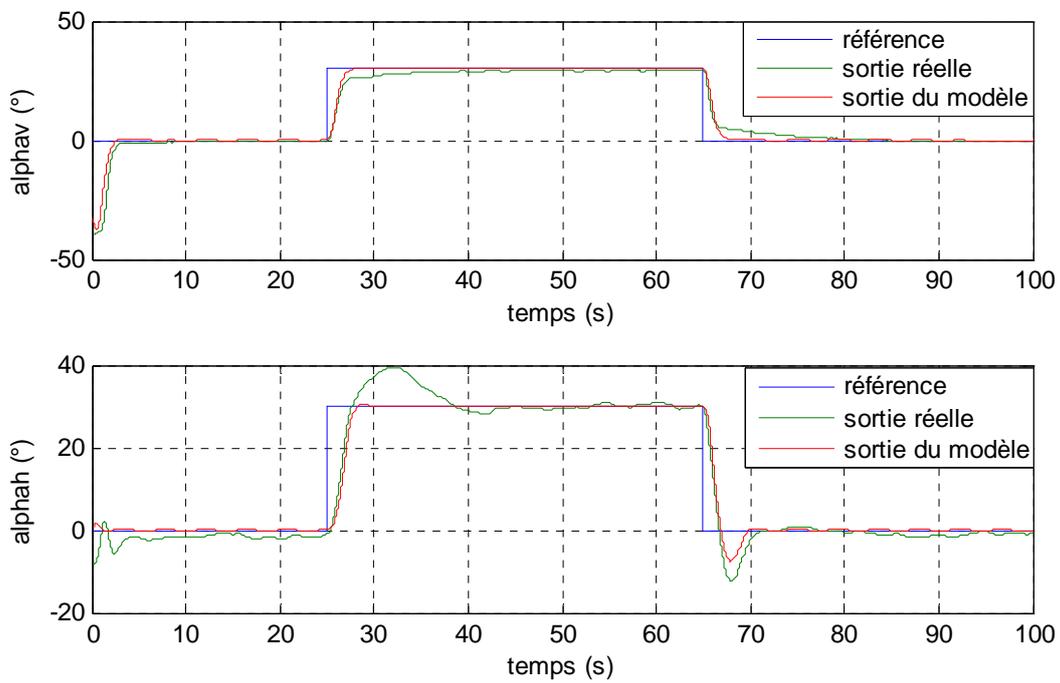


Figure VI.29 : réponse réelle avec réponse du modèle (commande RST flow)

## VI.6 Commande multi contrôleurs $H_\infty$ -RST

Afin de bénéficier des avantages de la commande  $H_\infty$  pour le sous système vertical, ainsi que des avantages de la commande  $RST$  pour le sous système horizontal, nous avons décidé de combiner leurs effets pour commander notre système dans l'espace. On obtient ainsi une commande hybride : commande  $H_\infty$  pour asservir l'angle  $\alpha_v$  et la commande  $RST$  pour asservir l'angle  $\alpha_h$ .

### VI.6.1 Commande multi contrôleurs $H_\infty$ -RST avec commutation franche

Les figures suivantes montrent les résultats expérimentaux obtenus avec cette commande couplée à un commutateur à événements discrets.

#### VI.6.1.1 Stabilisation et poursuite

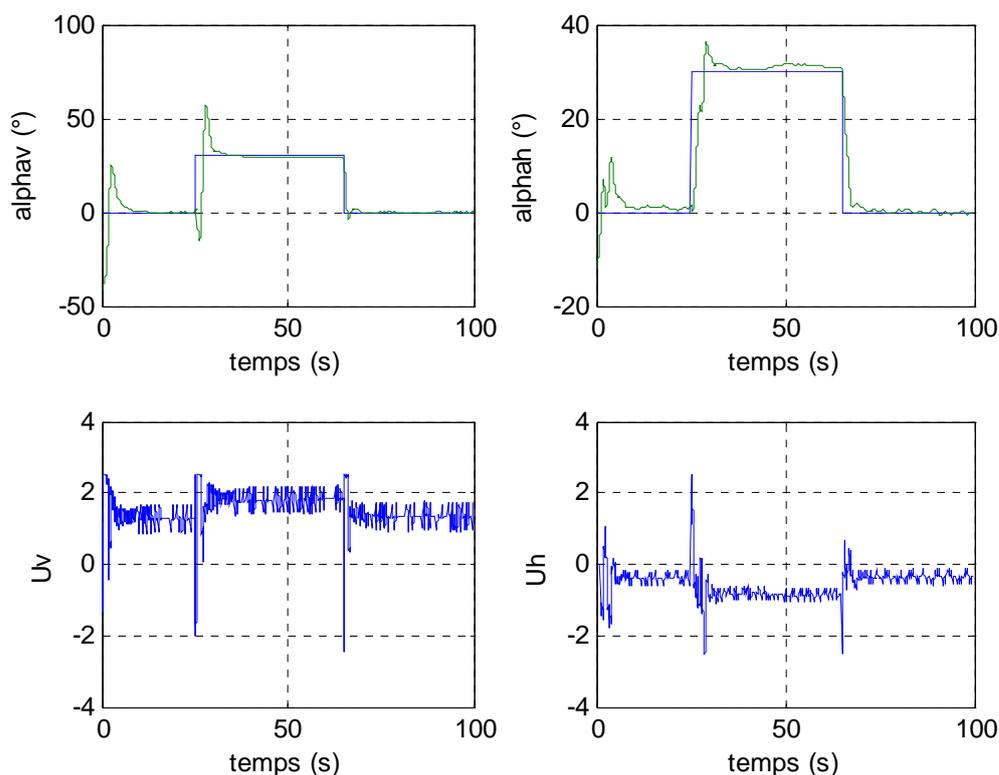


Figure VI.30 : stabilisation sur une référence carrée avec commande  $H_\infty$ -RST SED

La figure VI.30 illustre bien les améliorations apportées par cette commande hybride. On remarque qu'on a pu, avec la commande  $H_\infty$ , stabiliser l'angle d'élévation en un minimum de temps et pratiquement sans erreurs statiques, et, grâce à la commande  $RST$ , on a une réponse avec des dépassements réduits pour l'angle d'azimut.

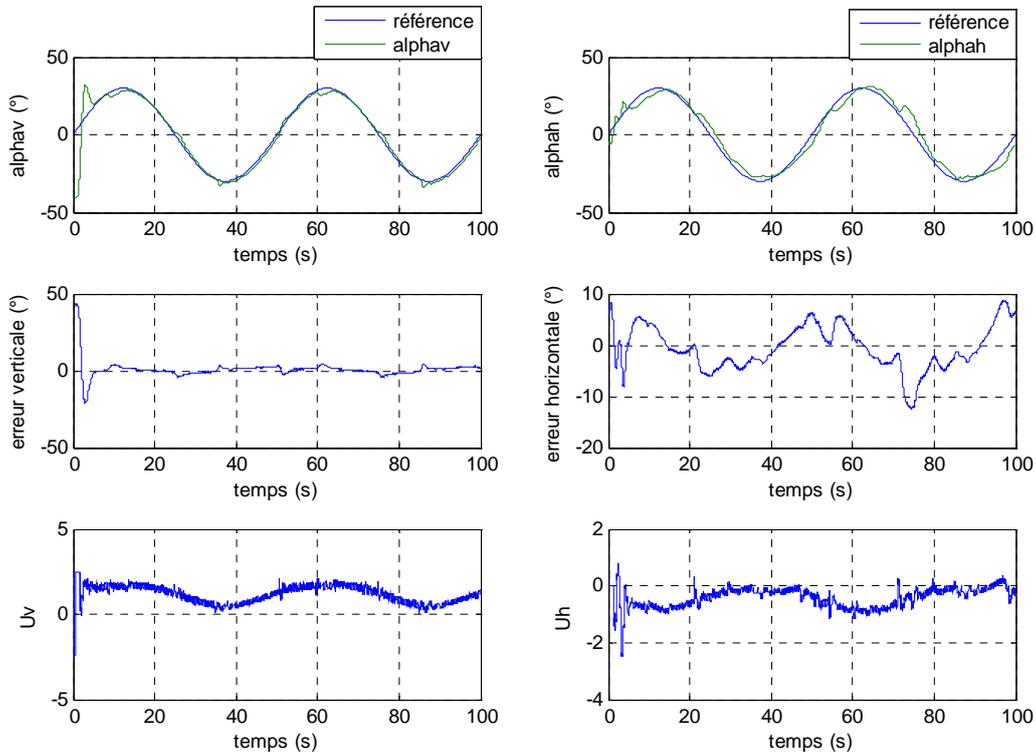


Figure VI.31 : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande  $H_\infty$ -RST CED

On remarque une diminution de l'erreur de poursuite. Cependant, des pics dans les commandes apparaissent engendrant de petits sauts dans les réponses du système. Ce problème sera résolu par la suite avec la commutation floue.

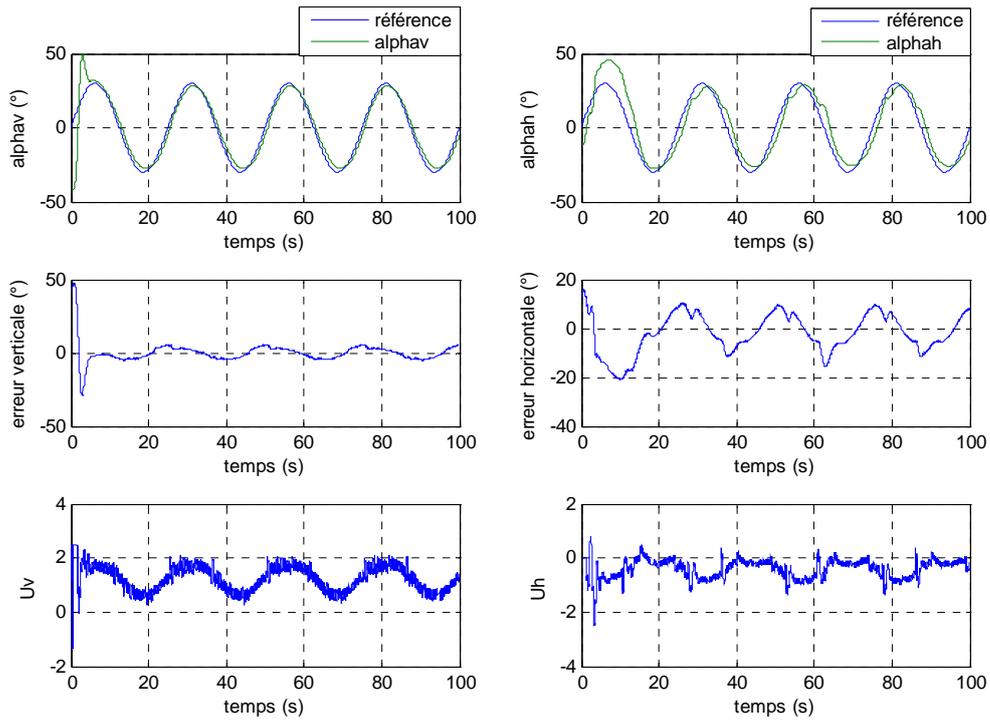


Figure VI.32 : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande  $H_{\infty}$ -RST SED

VI.6.1.2 Variations paramétriques

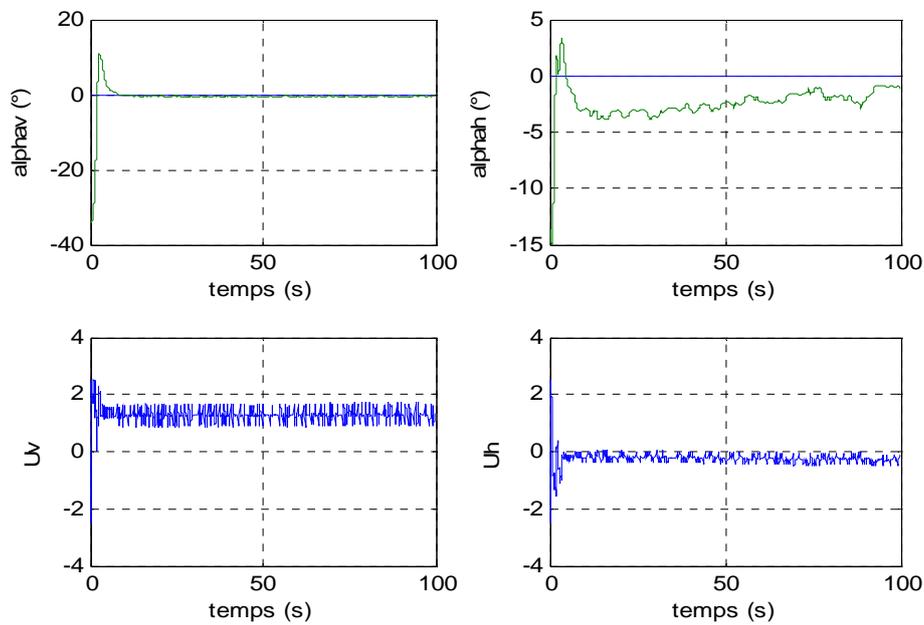


Figure VI.33 : stabilisation en présence de variations paramétriques avec commande  $H_{\infty}$ -RST SED

Les variations des paramètres du simulateur n'ont aucun effet sur le mouvement vertical car ce dernier est stabilisé par la commande  $H_\infty$  qui a démontré dans ce qui précède sa robustesse vis-à-vis les incertitudes du procédé. Par contre, les variations ont engendré une erreur de  $4^\circ$  maximum sur l'angle d'azimut, ce dernier est asservi par une commande  $RST$  qui est moins robuste que la commande  $H_\infty$ .

### VI.6.1.3 Rejet de perturbations

Les figures suivantes, montrent le bon rejet de perturbations et cela en un minimum de temps.

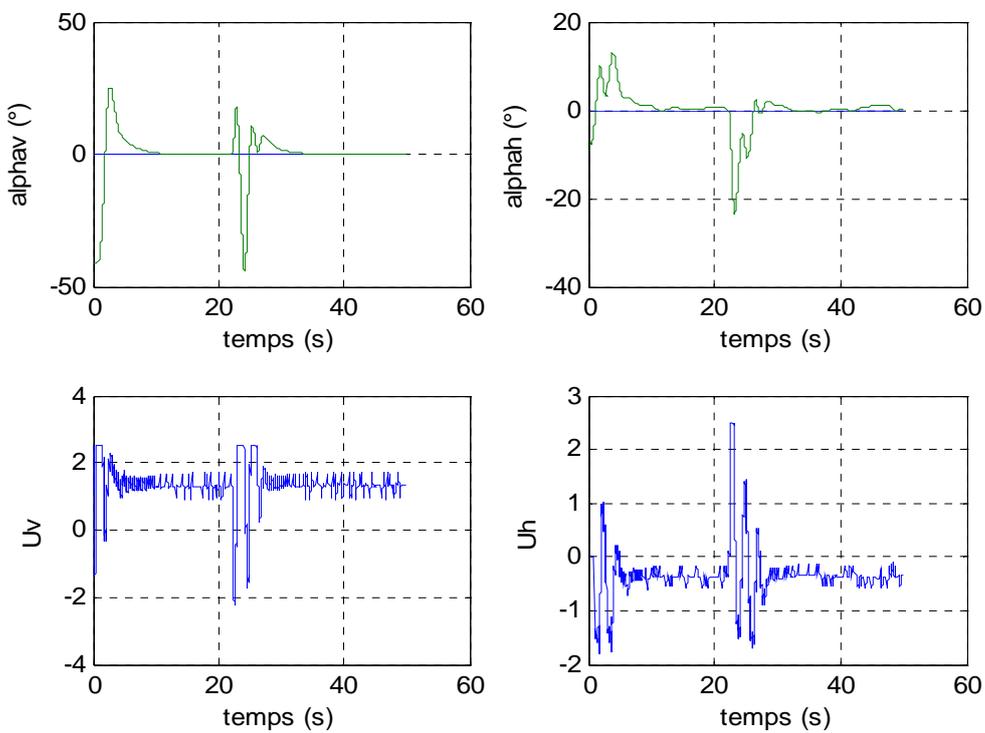


Figure VI.34 : rejet de perturbations par une commande  $H_\infty$ -RST CED

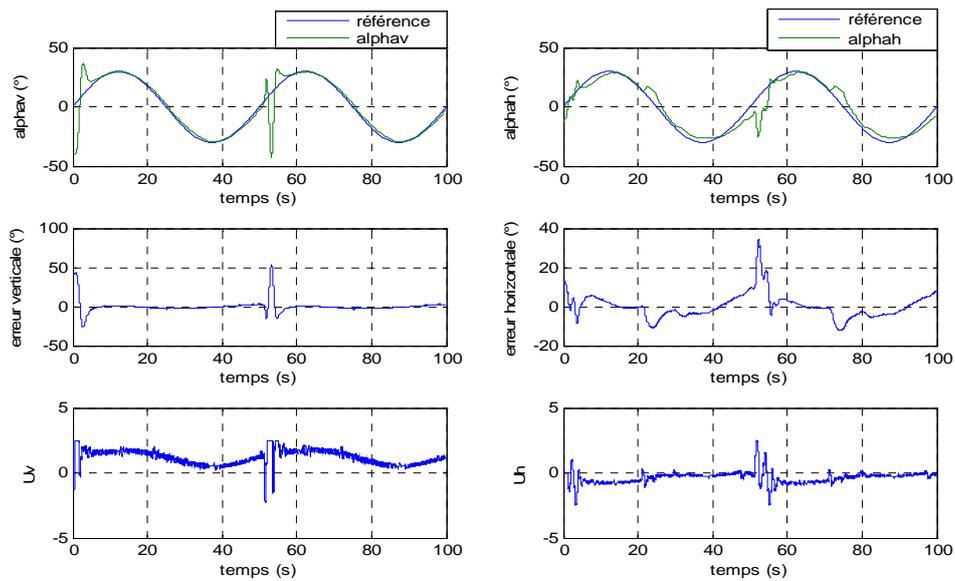


Figure VI.35 : rejet de perturbations par une commande  $H_{\infty}$ -RST CED lors d'une poursuite de sinusoïde

### VI.6.2 Commande multi contrôleurs $H_{\infty}$ -RST avec commutation floue

Pour éliminer les sauts dans les réponses du procédé, la commande développée précédemment a été couplée à un commutateur flou, et les résultats obtenus sont illustrés dans les figures ci-dessous.

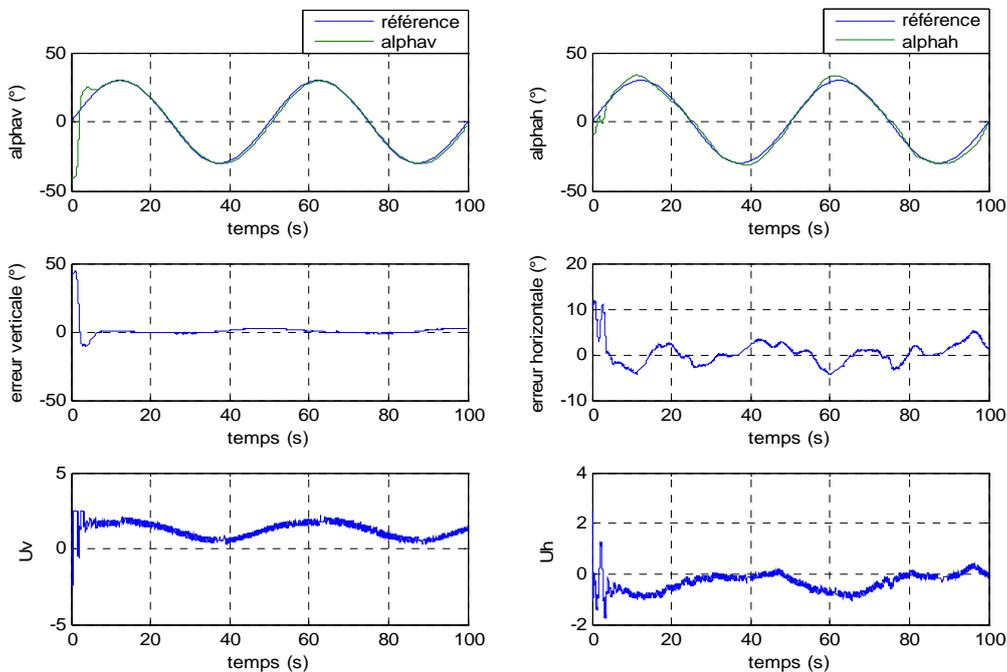


Figure VI.36 : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande  $H_{\infty}$ -RST floue

Les réponses précédentes montrent bien les améliorations apportées par la logique floue pour la commande multi contrôleurs. Elle a éliminé les discontinuités dans les signaux de commande, ce qui a permis d'avoir des réponses continues sans aucun saut, et une diminution sensible de l'erreur de poursuite est constatée.

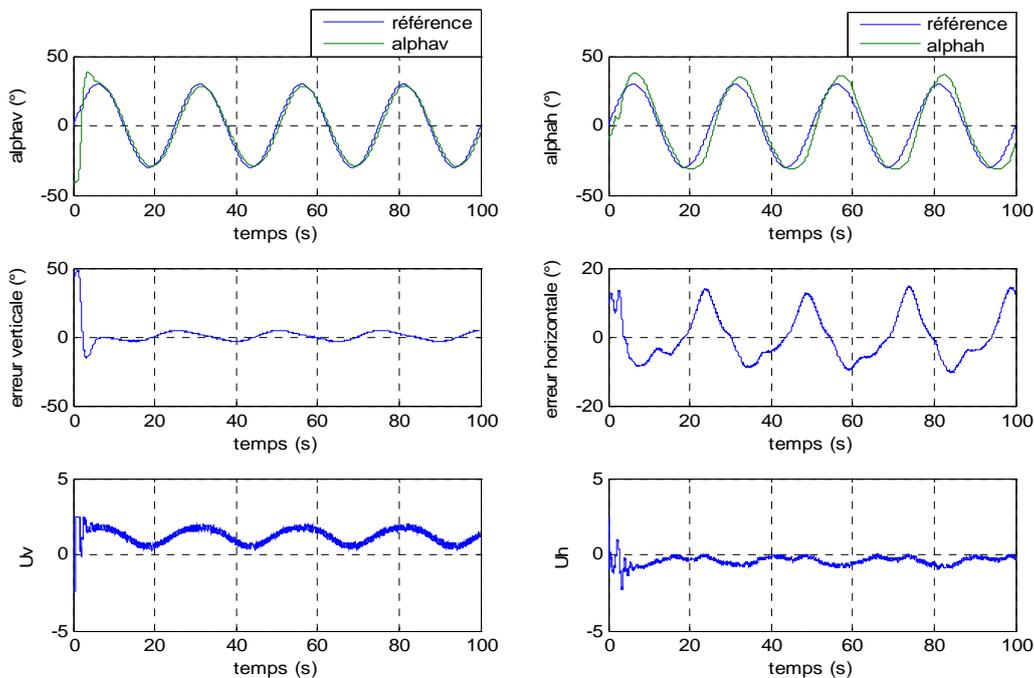


Figure VI.37 : poursuite d'une sinusoïde de fréquence 0.04 Hz avec commande  $H_{\infty}$ -RST floue

L'augmentation de la fréquence de la sinusoïde (dédoublment de la fréquence), a engendré une augmentation de l'erreur de poursuite, cela est dû essentiellement à l'inertie du simulateur d'hélicoptère qui est une structure mécanique assez lourde avec des actionneurs limités en puissance.

VI.6.2.1 Variations paramétriques

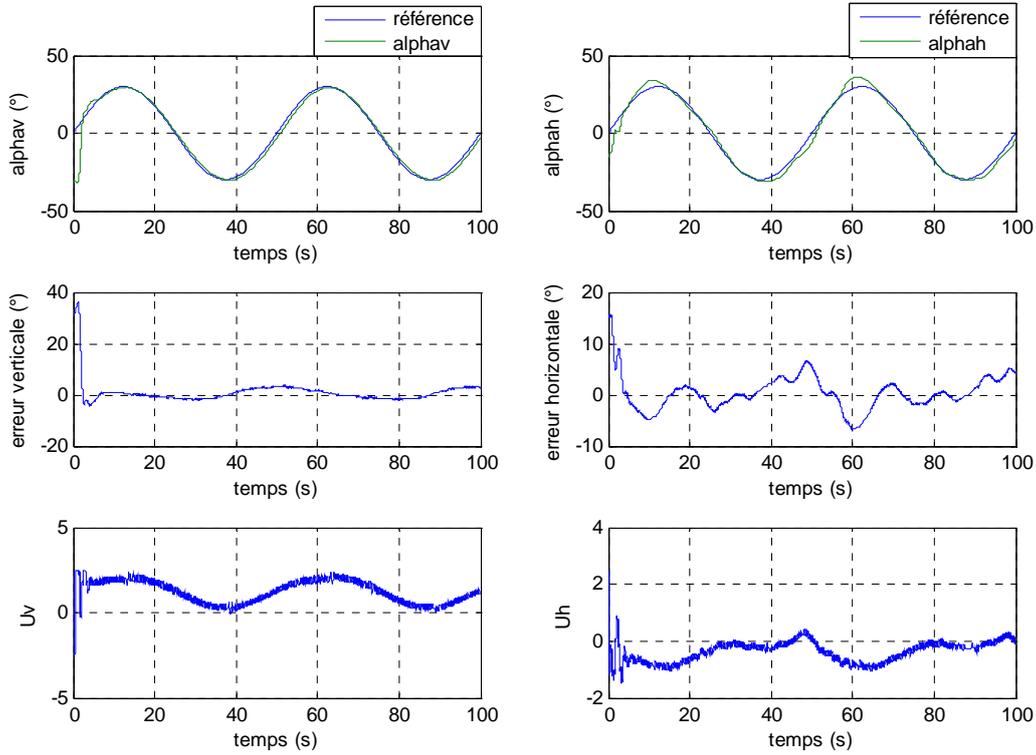


Figure VI.38 : poursuite d'une sinusoïde de fréquence 0.02 Hz avec commande  $H^\infty$ -RST floue en présence de variations paramétriques

Le commutateur flou a permis d'améliorer la robustesse de la commande appliquée comme le montrent la figure VI.38. En effet, le déplacement du centre de gravité de la structure a peu augmenté l'erreur de poursuite, elle est restée aux environs de  $\pm 2^\circ$  pour le vertical, et  $\pm 4^\circ$  pour l'horizontal avec quelques pics à  $\pm 6^\circ$ .

VI.7 Etude comparative

A fin de faciliter la comparaison entre les différentes lois de commandes, nous avons quantifié les résultats obtenus précédemment dans les tableaux suivants.

Tableau VI.1 : Comparaison des performances

		Commande multi contrôleurs $H_\infty$ à commutation franche	Commande multi contrôleurs RST à commutation franche	Commande multi contrôleurs $H_\infty$ - RST à commutation franche	Commande multi contrôleurs $H_\infty$ à commutation floue	Commande multi contrôleurs RST à commutation floue	Commande multi contrôleurs $H_\infty$ - RST à commutation floue
Temps de réponse (sec)	Vertical	7.270	4.593	7.660	6.459	7.485	6.677
	Horizontal	9.320	4.154	7.377	8.315	12.400	8.650
Maximum de l'erreur de poursuite (°)	Vertical	3.345	5.959	4.253	2.849	8.548	2.920
	Horizontal	8.130	10.800	12.370	6.573	10.148	5.138
Energie de l'erreur de poursuite (°)	Vertical	187.1587	536.572	269.5326	187.7196	$1.4273 \cdot 10^3$	156.805
	Horizontal	589.1646	$1.593 \cdot 10^3$	$1.5804 \cdot 10^3$	355.9493	$1.9926 \cdot 10^3$	364.5477
Energie de commande ( $v^2$ )	Vertical	178.175	178.175	178.175	178.175	178.175	178.171
	Horizontal	46.600	24.3921	29.416	46.600	23.455	25.921

Tableau VI.2 : Comparaison de la robustesse en présence de variations paramétriques

		Commande multi contrôleurs $H_\infty$ à commutation franche	Commande multi contrôleurs RST à commutation franche	Commande multi contrôleurs $H_\infty$ - RST à commutation franche	Commande multi contrôleurs $H_\infty$ à commutation floue	Commande multi contrôleurs RST à commutation floue	Commande multi contrôleurs $H_\infty$ - RST à commutation floue
Temps de réponse (sec)	Vertical	7.660	3.640	5.090	7.360	3.970	6.590
	Horizontal	9.150	3.850	7.315	9.050	5.060	8.915
Maximum de l'erreur de poursuite (°)	Vertical	4.2019	6.7751	6.253	3.7218	7.246	4.120
	Horizontal	5.1377	12.4973	13.370	5.0622	10.832	6.250

Tableau VI.3 : Comparaison des performances vis-à-vis du rejet de perturbation.

		Commande multi contrôleurs $H_\infty$ à commutation franche	Commande multi contrôleurs RST à commutation franche	Commande multi contrôleurs $H_\infty$ - RST à commutation franche	Commande multi contrôleurs $H_\infty$ à commutation floue	Commande multi contrôleurs RST à commutation floue	Commande multi contrôleurs $H_\infty$ - RST à commutation floue
Effet de perturbation (°)	Vertical	8.200	8.800	17.500	8.200	8.800	17.500
	Horizontal	-14.500	15.000	-23.600	-14.500	15.000	-23.600
Temps de rejet de perturbation (sec)	Vertical	5.850	5.710	7.5000	5.850	5.710	7.5000
	Vertical	7.140	3.500	5.7000	7.140	3.500	5.7000
Energie de commande ( $v^2$ )	Vertical	453.658	470.398	$1.7304 \cdot 10^3$	453.658	470.398	$1.7304 \cdot 10^3$
	Horizontal	234.457	257.922	628.5697	234.457	257.922	628.5697

Si on prend comme critère le temps de réponse du système, on constate que la commande multi contrôleurs RST à commutation franche présente la meilleure réponse. Cependant, si on considère le maximum de l'erreur de poursuite, cette commande n'est plus optimale. La meilleure commande (parmi celles étudiées) par rapport à ce dernier critère est la multi contrôleurs  $H_\infty$ -RST à commutation flou, qui, non seulement minimise l'erreur de poursuite (maximum et son intégral), mais consomme relativement peu d'énergie. Néanmoins, elle perd de ses performances en présence de variations paramétriques, comparé au  $H_\infty$  flou qui garde pratiquement les mêmes performances. Cela est dû à la robustesse de ses contrôleurs ( $H_\infty$ ).

La commutation par logique flou a globalement amélioré les performances des commandes multi contrôleurs, sauf pour le RST où elle a retardé la réponse et augmenté les erreurs, mais a tout de même atténué les sauts dans les réponses.

Le rejet de perturbation est assuré par toutes les commandes en un temps proportionnel à son intensité et au temps de réponse du système.

## VI.8 Conclusion

Dans ce chapitre, nous avons présenté les résultats expérimentaux des différentes lois de commandes multi contrôleurs. Nous avons vu que la commande multi contrôleurs RST présente un temps de réponse assez court, mais n'a pas un bon suivi de consignes variables. La commande multi contrôleurs  $H_\infty$ -RST présente de bonnes performances, mais elles diminuent en présence de variations paramétriques. La commande multi contrôleurs  $H_\infty$  n'est ni la plus rapide, ni la plus performance mais assure une bonne robustesse. La commutation floue a pu améliorer les performances des différentes commandes citées ci-dessus, sauf le RST où elles ont été réduites. Toutefois, elle a permis d'éliminer les sauts dus à la commutation franche, tout en augmentant la robustesse de la commande multi contrôleurs RST.

## Conclusion générale

L'objectif de ce mémoire était de dimensionner une commande multi-contrôleurs pour l'asservissement des angles d'élévation et d'azimut du simulateur d'hélicoptère TRMS. Pour cela, différentes approches de commande ont été présentées.

En effet, la modélisation du TRMS nous a conduits à un modèle mathématique multivariable, non linéaire et fortement couplé, où une commande linéaire autour d'un seul point donnerait des performances non satisfaisantes (exemple du PID).

L'avantage de la commande multi-contrôleur réside dans le fait que les régulateurs locaux peuvent être de natures différentes. Dans le cadre de ce travail, des régulateurs locaux  $H_\infty$  ainsi que des régulateurs  $RST$  sont synthétisés. La commande finale appliquée au système résulte de la sélection d'un ou de plusieurs régulateurs adéquats selon la nature de la commutation.

Les simulations effectuées sur le procédé avec la commande multi-contrôleur ont mis en évidence l'inconvénient majeur de cette commande lorsque celle-ci utilise un commutateur « tout ou rien ». Cette commutation franche introduit des discontinuités au niveau des commandes qui engendrent des sauts dans les réponses du système d'où de fortes sollicitations des actionneurs.

Pour palier à ce problème, des solutions ont été proposées pour diminuer ces sauts. Cependant, la meilleure solution reste la logique floue, celle-ci rend le signal de commande doux et continu car le passage d'un régulateur à un autre se fait progressivement.

Afin de valider notre travail, l'implémentation pratique a été effectuée sur le simulateur d'hélicoptère. Les résultats expérimentaux montrent l'efficacité de cette commande pour la

poursuite de trajectoire et la stabilisation en dépit de la présence de phénomènes non modélisés, comme la force de torsion des câbles et quelques paramètres incertains telle les forces aérodynamiques des hélices et les caractéristiques non linéaire des propulseurs. Ceci montre la robustesse de ces lois de commande.

De plus, l'absence de gravité dans le mouvement horizontal, rend celui-ci difficile à stabiliser à cause des oscillations non compensées. Contrairement à l'horizontal, le mouvement vertical est bien amortie grâce à l'effet de la pesanteur. Celle-ci contribue à l'atténuation des oscillations.

Les problèmes d'ordre pratique, nous ont amenés à apporter quelques corrections aux commandes synthétisées, particulièrement à la commande *RST*, où l'ajout de quelques termes constants en «feedforward» s'avère nécessaire pour diminuer les erreurs de trainage.

Les derniers essais expérimentaux, ont mis en évidence l'intérêt de la commande multi-contrôleurs, qui permet d'améliorer les performances en mettant en œuvre des contrôleurs locaux de natures différentes (commande  $H_\infty$  - *RST*). Cette dernière solution, a permis d'avoir de meilleures performances que celles obtenues avec des commandes multi-contrôleurs de même nature sur les deux axes.

## Bibliographie

- [1] F. Farah, F. Hachid, « *Synthèse et implémentation pratique de lois de commandes non linéaires : Application à un simulateur d'hélicoptère,* » projet de fin d'études, Ecole Nationale Polytechnique d'Alger, juin 2005.
- [2] X.D. Koutsoukos P.J. Antsaklis, “*Hybrid Control Systems Using Timed Petri Nets: Supervisory Control Design Based on invariant Properties,*” Department of Electrical Engineering, University of Notre Dame, 1999.
- [3] *Manual of the Twin Rotor MIMO System “Installation and Commissioning,”* manual 33-948\_949-0V61.
- [4] *Manual of the Twin Rotor MIMO System “Control Experiment,”* manual 33-948\_949-1V61.
- [5] *Manual of the Twin Rotor MIMO System “Environment introduction and software installation,”* manual 33-000V61.
- [6] *Manual of the Twin Rotor MIMO System “advanced teaching manual,”* manual 330074M5.
- [7] O. Pages, « *Etude et comparaison de différentes structures de commande multi-contrôleurs. Application à un axe robotise,* » Thèse de doctorat, Université de Savoie, Juillet 2001.

- [8] Manual, Getting Started with Real-Time Workshop For Use with SIMULINK, American: MathWork Inc., 2002–2005 .
- [9] J. Gucknheimer, “*A robust hybrid stabilisation strategy for equilibria*,” center for applied mathematics, cornell university, Ithaca, New York, U.S, 1995
- [10] Y. Piguet, « *synthèse multi-modèle d'un régulateur polynomial robuste*, » Thèse de Doctorat des sciences, Ecole Polytechnique Fédérale de Lausanne, 1997
- [11] R. Longchamp. « *Commande numérique des systèmes dynamiques*, » Thèse de Doctorat, presses polytechniques et universitaires romandes, Lausanne, 1995.
- [12] M. El-Khoury. « *Influence des zéros d'une fonction de transfert sur le comportement dynamique d'un système linéaire et application au réglage polynomial*, » Thèse de doctorat, Institut d'automatique, EPFL, Lausanne, 1991.
- [13] A. Martini, F. Leonard et G. Abba, « *suivi robuste de trajectoires d'un hélicoptère drone sous rafale de vent*, » laboratoire de génie Industriel et production mécanique (LGIPM). Ecole Nationale d'Ingénieurs de Metz, 2006.
- [14] P. Chevrel. « *Commande multi-contrôleurs : quelles application*», Conférence, journée « multi-contrôleurs, » organisée par le GR Robuste, Supelec, 2000.
- [15] B. Mansouri, « *Contribution à la synthèse de lois de commandes en poursuite de trajectoire pour les systèmes flous de type takagi sugeno Incertains*, » thèse de doctorat en génie informatique, automatique et traitement du signal, Université de Reims Champagne Ardenne, 2005.
- [16] M. Rivoire, J-L. Ferrier, « *Matlab, Simulink, Stateflow*, » Editions TECHNIP, Université d'Angers, 2001.
- [17] L. Baghli, « *Contribution à la commande de la machine asynchrone, utilisation de la logique floue, des réseaux de neurones et des algorithmes génétiques*, » thèse de doctorat en génie électrique, université Henri Poincaré, Nancy-I.

- [18] A. Noth, « *Synthèse et implémentation d'un contrôleur pour micro Hélicoptère à 4 Rotors,* » projet de diplôme en microélectronique, Ecole Polytechnique Fédérale de Lausanne, 2004.
- [19] M. Rodrigues « *Diagnostic et commande active tolérante aux défauts appliqués aux systèmes décrits par des multi-modèles linéaires,* » thèse de doctorat en automatique, université Henri pointcarré, Nancy, 2005.
- [20] A.N. Bennamia, A. Salamani, « *Commandes robustes appliquées à la stabilisation d'un Navire,* » projet de fin d'études en automatique, Ecole Nationale Polytechnique d'Alger, 2007.
- [21] S. Font, G. Duc, F. Carrère, « *Commande fréquentielle robuste : application aux paliers magnétiques,* » Technique de l'ingénieur, R7432.
- [22] F. Gigon, « *Contribution à la conception d'un micro hélicoptère à rotors coaxiaux,* » projet de semestre en microtechnique, Ecole Polytechnique Fédérale de Lausanne, 2005.
- [23] K. Alireza, « *Robuste control,* » cours d'automatique, Ecole Polytechnique Fédérale de Lausanne.
- [24] K. Alireza, « *Robuste controler design by convex optimisation,* » cours d'automatique, école polytechnique fédérale de Lausanne.
- [25] Rahideh, A., et al. "Dynamic modelling of a TRMS using analytical and empirical approaches," Control Engineering Practice, 2007.
- [26] S.M. Ahmad, A.J.Chipperfield, M.O. Tohki, "Modelling and control of a Twin Rotor Multi-input Multi output System," American Control Conference, Chicago, Illinois, juin 2000.

- [27] S.M. Ahmad, M.H.Shaheed, A.J.Chipperfield et M.O. Tokhi, “*Nonlinear modeling of a twin rotor MIMO system using radial basis function networks,*” Department of Automatic Control and Systems Engineering, The University of Sheffield, 2000.
- [28] S.M. Ahmad, A.J.Chipperfield, M.O. Tohki, “*Dynamic modeling and optimal control of a twin rotor MIMO system,*” Department of Automatic Control and Systems Engineering, The University of Sheffield, 2000.
- [29] S.M. Ahmad, A.J.Chipperfield, M.O. Tohki, “*Dynamic modeling and control of a 2 DOF MIMO system,*” 2000, pp.1451-1456.
- [30] Juhng-Perng Su, Chi-Ying Liang and Hung-Ming Chen, “*Robust Control of a class of nonlinear systems and its application to a Twin Rotor MIMO System,*” IEEE ICIT 2002, Bangkok, THAILAND, pp. 1272-1277.
- [31] Badar UL Islam, Nisar Ahmed, Daud Latif, Bhatti, Shahid Khan, “*Controller design using Fuzzy Logic For a Twin Rotor MIMO system,*” Proceedings IEEE INMIC, 2003, pp.264-268.
- [32] Jih-Gau Juang, Wen-Kai Liu, Cheng-Yu Tsai, “*Intelligent control scheme for Twin Rotor MIMO System,*” proceedings on the IEEE International Conference on Mechatronics, 10-12 juillet 2005, Taipei, Taiwan, pp. 102-107.
- [33] Chuan-Sheng Liu, Liang-Rui Chen, Bieng-Ze Li, Shih-Kai Chen, Zhao-Syong Zeng, “*Improvement of the Twin Rotor MIMO System tracking and transient response using Fuzzy control Technology,*” ICIEA 2006.
- [34] A. Rahideh, M.H. Shaheed, “*Hybrid Fuzzy PID based control of a Twin Rotor MIMO System,*” 2006, pp. 49-54.
- [35] Rahideh, M.H. Shaheed, A.H. Bajodah, “*Adaptative nonlinear model inversion of a Twin Rotor MIMO System using artificial intelligence,*” 16ème IEEE International Conference on Control Applications, Part of IEEE Multi-conference on Systems and Control Singapore, TuC03.5, 1-3 Octobre 2007, PP 898-903.

- [36] Te-Wei Lu, P. Wen, “*Time optimal and Robust Control of Twin Rotor System,*” IEEE International Conference on Control and Automation, Guangzhou, CHINA, ThA3-2, 30 Mai -1 Juin, 2007, pp 862-866.
- [37] Jonas Balderud, “*Modelling and Control of a Toy-Helicopter,*” Department of Electrical Engineering, master’s thesis, January 2002

## Annexe A

## Carte d'acquisition : ADVANTECH PCI 1711

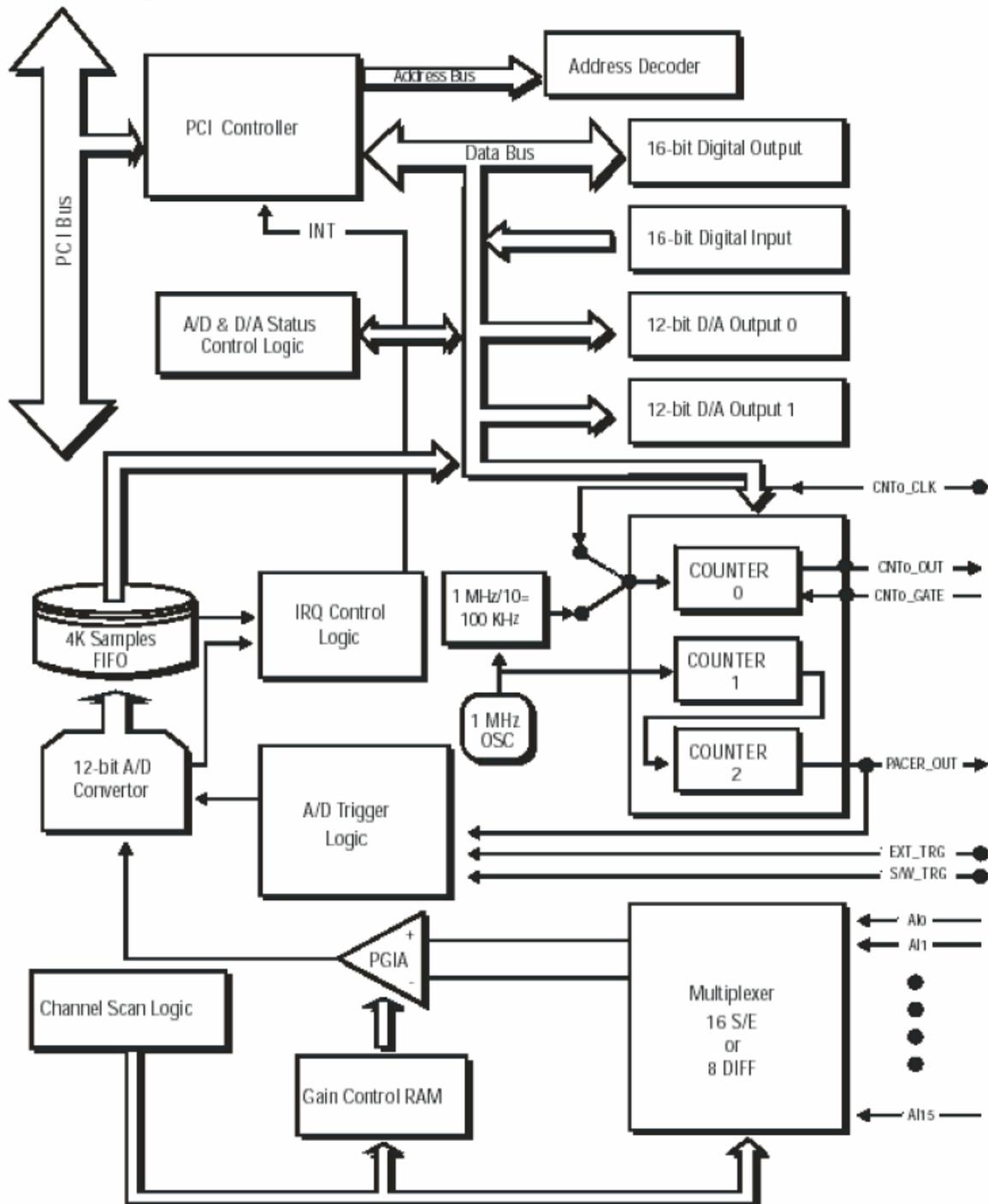
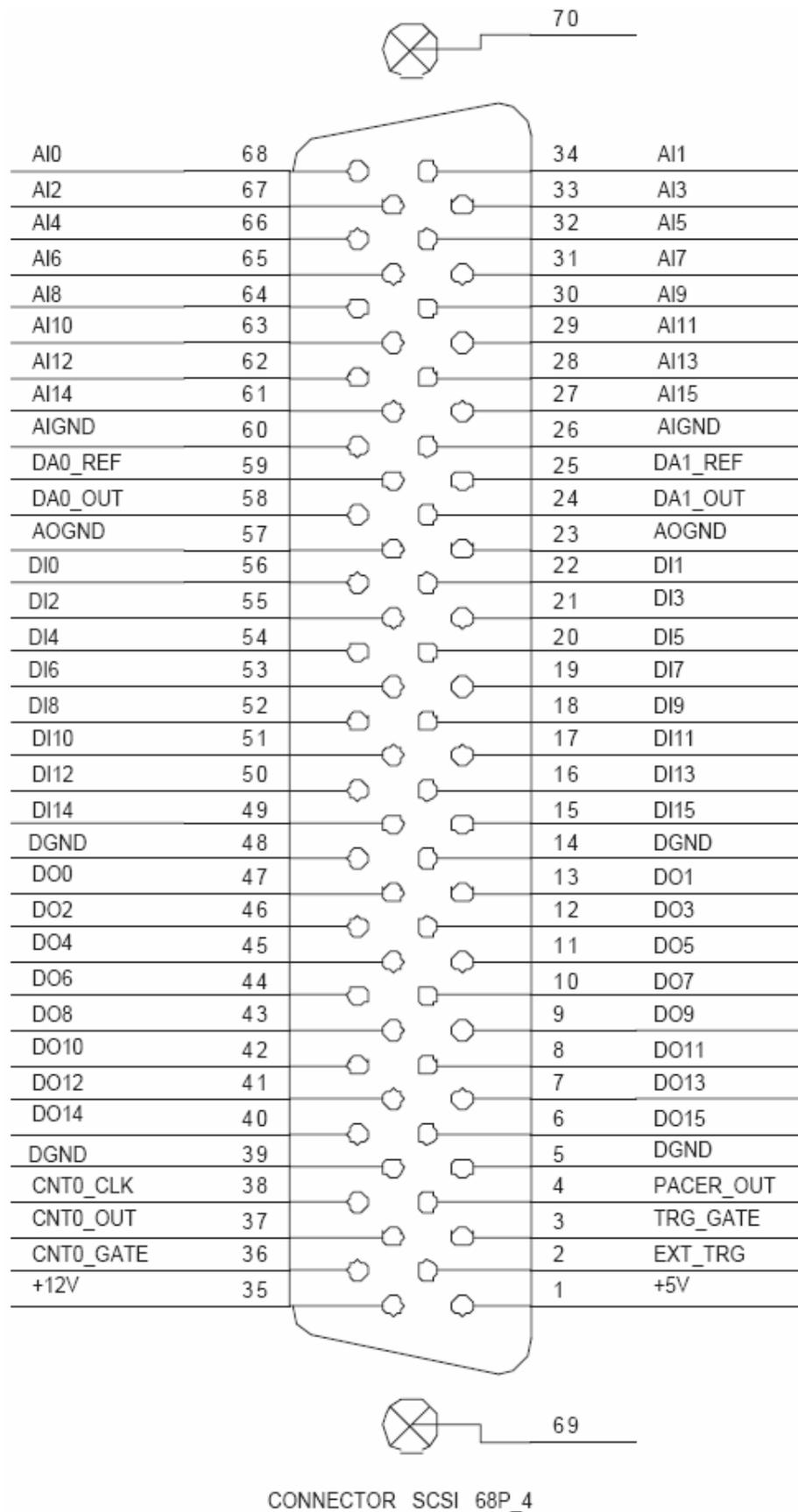


Figure A.1 : diagramme en blocks



**Figure A.2 :** attribution des pins du connecteur E/S du PCI-1711

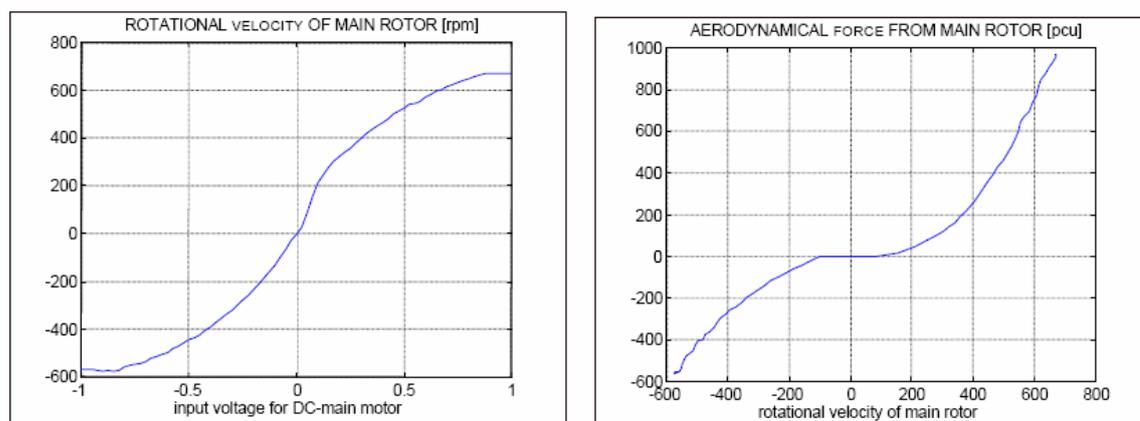
Tableau A.1 : description des différents signaux de connexion entrée sortie

Nom du signal	Référence	Direction	Description
AI<0...15>	AIGND	Entrée	<b>Canal d'entrée analogique 0 jusqu'à 15.</b> Chaque paire de canal, AI <i, i+1> , (i=0,1,2,...,14) peut être configuré aussi bien en single-ended qu'en entrées différentielles de la carte PCI-1711
AIGND	-	-	<b>Entrée analogique de masse (ground).</b> Les trois références de terre (AIGND, AOGND et DGND) sont connectées ensemble sur la carte PCI-1711
AO0_REF AO1_REF	AOGND	Entrée	<b>Canal de sortie analogique 0/1 de référence extérieure</b>
AO0_OUT AO1_OUT	AOGND	Sortie	<b>Canaux de sortie analogique 0/1</b>
AOGND	-	-	<b>Sortie analogique de masse (Ground).</b> Les tensions analogiques de sortie ont cette pin comme référence. Les trois références de terre (AIGND, AOGND et DGND) sont connectées ensemble sur la carte PCI-1711
DI<0...15>	DGND	Entrée	<b>Canaux d'entrée digitale</b>
DO<0...15>	DGND	Sortie	<b>Canaux de sortie digitale</b>
DGND	-	-	<b>Masse digitale,</b> cette pin fournit la référence pour les canaux digitaux aux connecteurs E/S tout comme pour l'alimentation continue + 5V. Les trois références de terre (AIGND, AOGND et DGND) sont connectées ensemble sur la carte PCI-1711
CNT0_CLK	DGND	Entrée	<b>Entrée horloge du compteur 0.</b> L'entrée horloge du compteur peut aussi bien être interne qu'externe, cela se détermine par programme.
CNT0_OUT	DGND	Sortie	Sortie du compteur 0
CNT0_GATE	DGND	Entrée	Control du pont compteur 0
PACER_OUT	DGND	Sortie	<b>Sortie impulsion horloge.</b> Cette pin donne une impulsion à chaque fois qu'une impulsion horloge se met à 1. Si le convertisseur A/N est mis sur l'état trigger cet sortie peut être utilisée comme signal de synchronisation pour d'autres applications. Un front montant détermine le début de la conversion
TRG_GATE	DGND	Entrée	<b>Porte trigger externe A/D.</b> quand TRG_GATE est connecté à +5V, il définit le signal trigger comme une entrée. Quand TRG_GATE est connecté à DGND, il désélectionne le signal trigger, et il n'est plus en entrée
EXT_TRG	DGND	Entrée	<b>Trigger externe A/D.</b> cette pin est un signal d'entrée trigger externe pour la conversion A/N. un front montant active la conversion A/N
+12V	DGND	Sortie	<b>Source 12V continue</b>
+5V	DGND	Sortie	<b>Source 5V continue</b>

## Annexe B

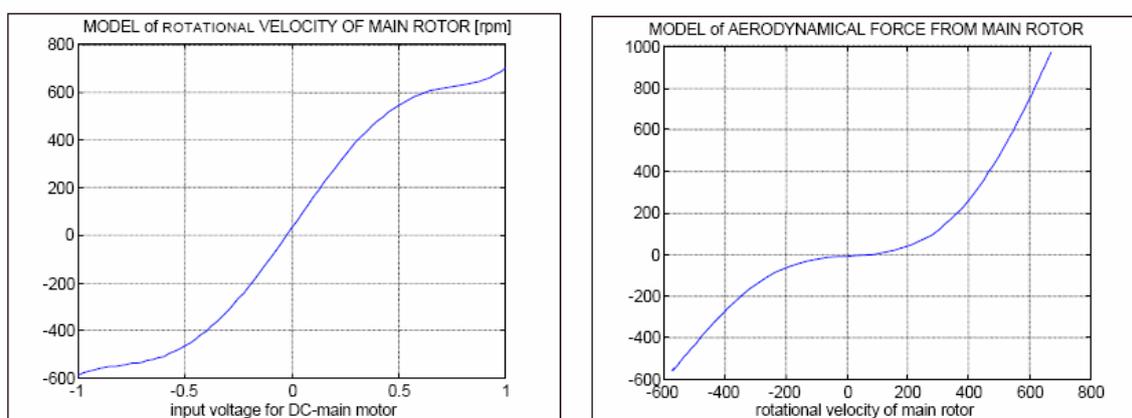
### B.1 Courbe expérimentales des caractéristiques non linéaires [6]

On obtient expérimentalement les deux courbes de la vitesse de rotation des rotors en fonction de la tension d'alimentation des moteurs (principal et secondaire).



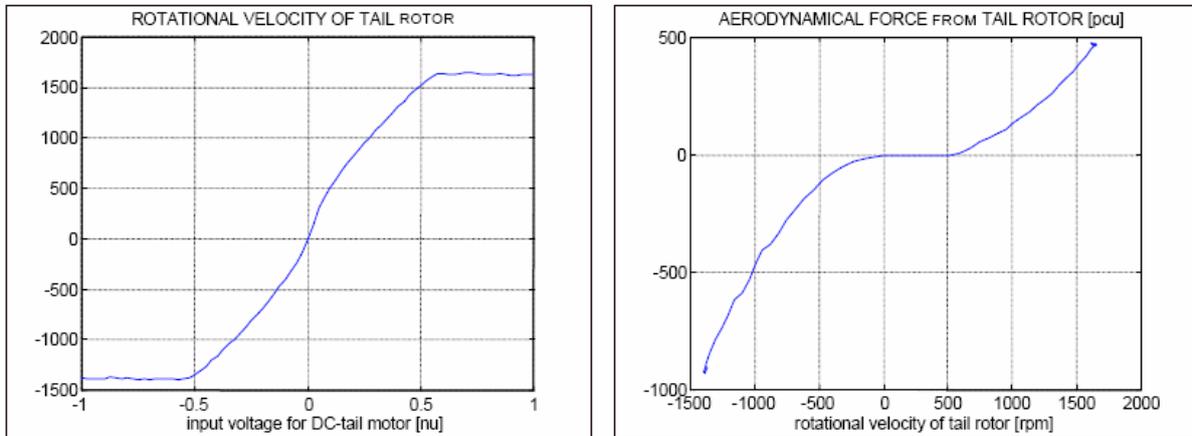
**Figure B.1 :** graphes expérimentaux représentant la vitesse de rotation du rotor principal en fonction de la tension à l'entrée du rotor (à gauche) et la force aérodynamique engendrée par la rotation du rotor principal (à droite)

L'utilisation de la fonction *polyfit* de Matlab, nous permet d'approximer ces courbes par des polynômes déjà donnés au chapitre II, le tracé de ces polynômes est illustré par les figures ci-dessous :

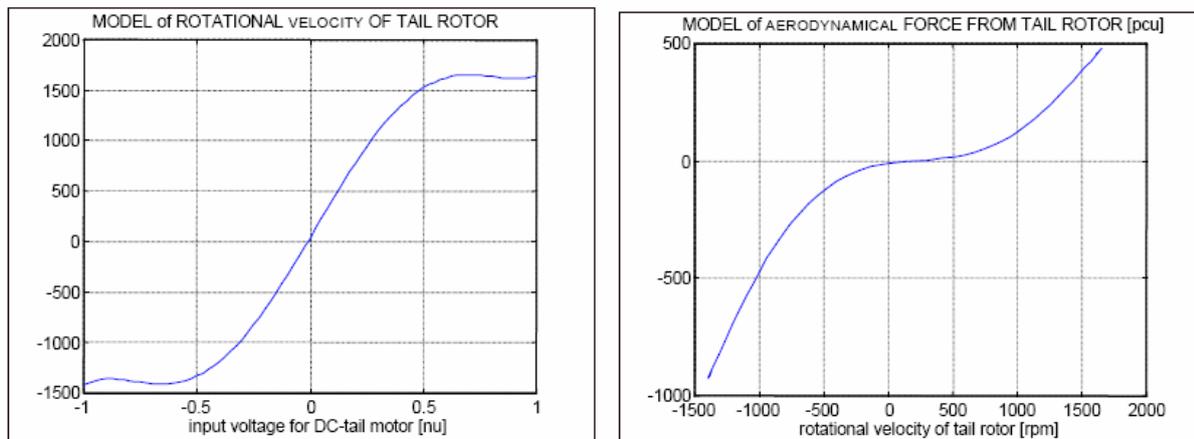


**Figure B.2 :** approximation polynomiale des caractéristiques du rotor principal

De même, les courbes expérimentales des forces aérodynamiques en fonction des vitesses de rotation des rotors, sont données par les figures suivantes :



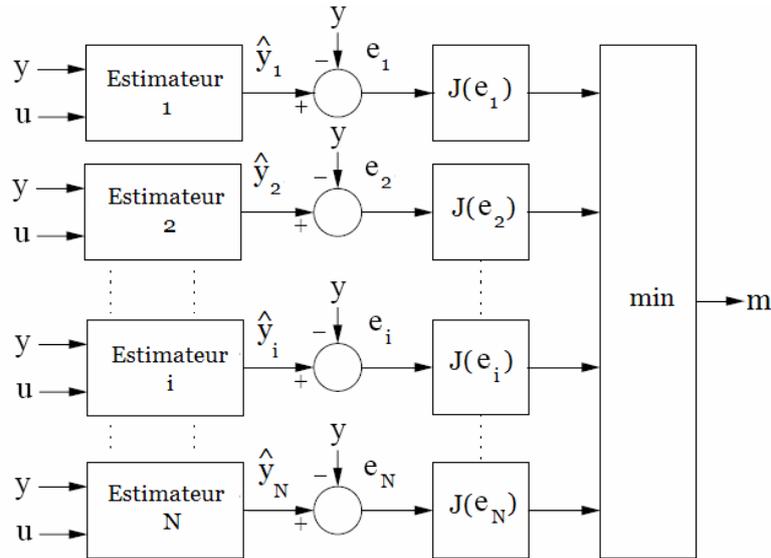
**Figure B.3 :** graphes expérimentaux représentant la vitesse de rotation du rotor secondaire en fonction de la tension à l'entrée du rotor (à gauche) et la force aérodynamique engendrée par la rotation du rotor secondaire (à droite)



**Figure B.4 :** approximation polynomiale des caractéristiques du rotor de queue

## B.2 Commande multi contrôleurs : approche indirecte

La structure du superviseur est illustrée par la figure suivante :



**Figure B.5 :** Structure du superviseur basée sur la prédiction et l'optimisation d'un critère

L'idée de la méthode selon [pages 01] est d'associer un prédicteur (ou un modèle) et un contrôleur à chaque environnement spécifique. Un nombre fini de prédicteurs sera donc utilisé pour décrire l'ensemble des environnements, ce qui constituera le superviseur. Un algorithme de commutation basé sur la minimisation d'un critère quadratique permettra de connaître rapidement le "meilleur" prédicteur et donc d'appliquer au procédé la sortie du contrôleur correspondant. Le critère utilisé par l'algorithme de commutation s'écrit :

$$J_i(t) = \alpha \cdot e_i^2(t) + \beta \cdot \int_0^t \exp(-\gamma \cdot (t - \tau)) e_i^2(\tau) \cdot d\tau, \quad (\text{B.1})$$

Avec :  $\alpha, \beta, \gamma \geq 0$

$$e_i(t) = \hat{y}_i - y$$

L'équation (III.1) est fonction de la puissance instantanée de l'erreur d'estimation et de la fonction d'inter-corrélation entre cette puissance et une fonction du temps exponentielle permettant d'introduire un facteur d'oubli. On note la puissance instantanée du signal d'erreur par :

$$P_i(t) = e_i^2(t) \quad (\text{B.2})$$

En passant dans le domaine fréquentiel par la transformée de Laplace, la fonction de transfert entre le critère et la puissance de l'erreur d'estimation est :

$$\frac{J_i(s)}{P_i(s)} = \alpha + \frac{\beta}{s + \gamma} \quad (\text{B.3})$$

Cette fonction de transfert est en fait un filtre passe-bas dont les paramètres permettent de limiter la dynamique d'évolution des critères  $J_i$ .

Au critère  $J_i$  le plus faible correspond, donc, le modèle  $M_i$  le plus représentatif du procédé à l'instant  $t$ . Soit, s'il y a  $N$  modèles du procédé, le modèle le plus représentatif est le modèle  $M_m$  où :

$$m(t) = \min_i \{J_i(t), i \in [1, N]\} \quad (\text{B.4})$$

Le contrôleur utilisé est alors le contrôleur  $C_m$  parmi les  $N$  contrôleurs associés respectivement aux  $N$  modèles.

L'architecture de cette approche peut être résumée par la figure suivante :

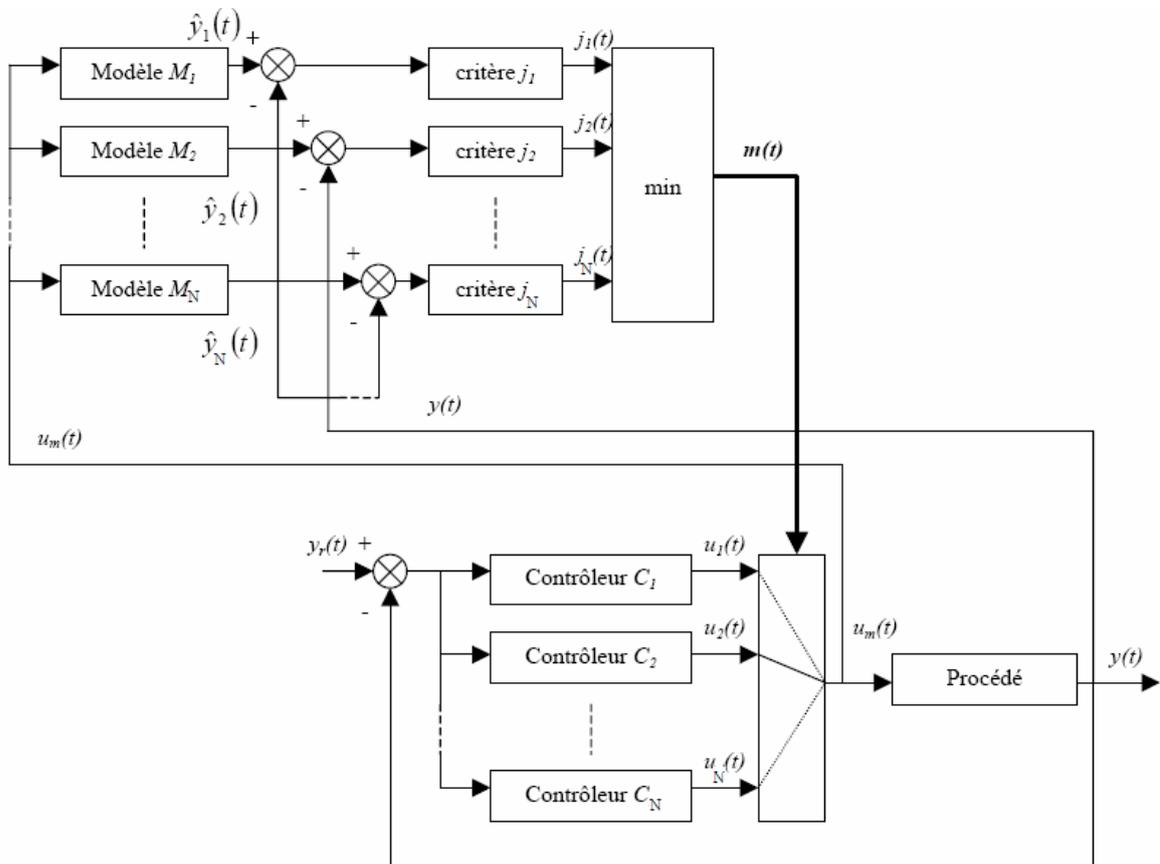


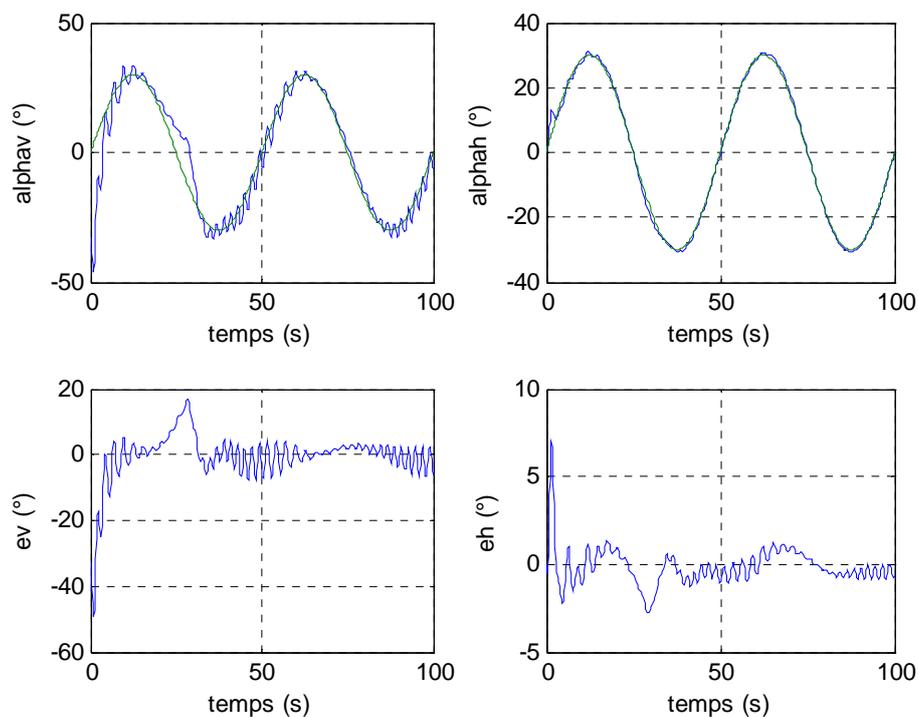
Figure B.6 : schéma globale d'une commande multi contrôleurs supervisée

Les trois paramètres :  $\alpha$ ,  $\beta$  et  $\gamma$  jouent un rôle important : au début de l'adaptation, seul  $e_i^2(t)$  qui correspond à des valeurs instantanées intervient dans l'équation (III.1), mais cette erreur va tendre vers zéro et le terme intégral qui fait la somme des valeurs passées deviendra peu à peu prépondérant. En fait,  $e_i^2(t)$  est une mesure de l'instantané tandis que le terme intégral prend en compte les valeurs passées. De ce fait, un compromis doit être fait entre une mesure instantanée pondérée par  $\alpha$  et une mesure à long terme pondérée par  $\beta$ .

$\gamma$  est un facteur d'oubli : il pondère les valeurs passées. De plus, une valeur non nulle de  $\gamma$  assure la bornitude de l'intégrale (B.1).

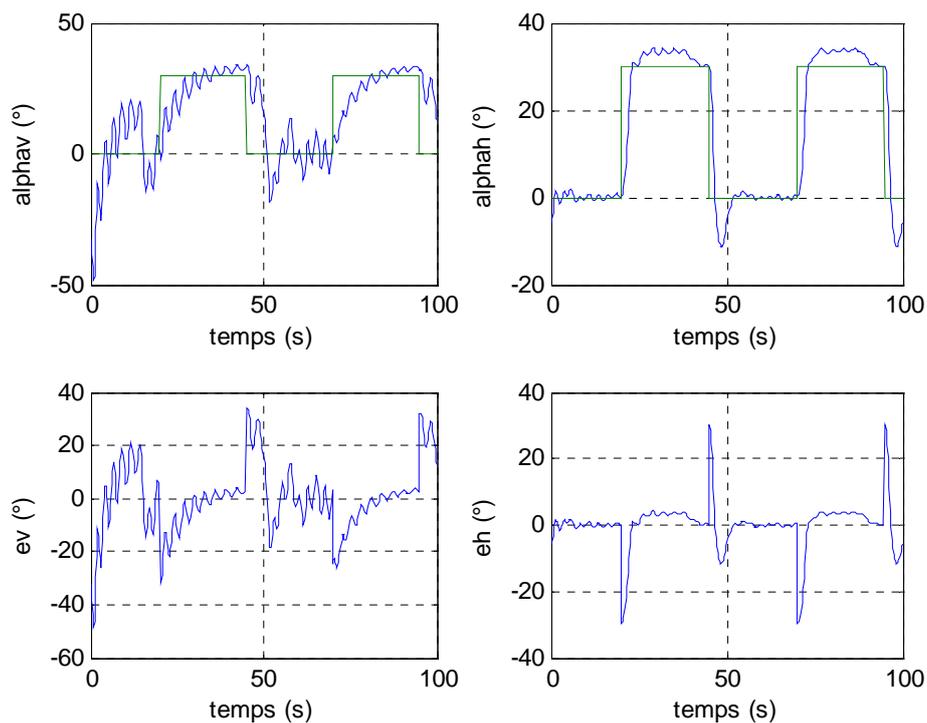
## Annexe C

### Réponse du PID



**Figure C.1 :** Réponse du régulateur PID pour une poursuite de sinusoïde de fréquence 0.02 Hz et d'amplitude  $30^{\circ}$

Il y a une bonne poursuite de sinusoïde pour l'angle d'azimut, contrairement à l'angle d'élévation qui présente beaucoup d'oscillations et un temps de réponse long.



**Figure C.2 :** Réponse du régulateur PID pour une poursuite d'un signal en créneaux de fréquence 0.02 Hz et d'amplitude  $30^\circ$

La lenteur de la réponse du tangage se fait sentir dans la poursuite du signal en créneaux, de plus celle-ci est très oscillatoire. Pour l'horizontal, la réponse est plus rapide, mais l'erreur statique, disparaît après un temps assez grand.

Le PID ne présente pas de bonnes réponses en poursuite. Néanmoins, en pratique, les oscillations sont filtrées par le système donnant, ainsi, des réponses plus lisses.

Le PID donne des résultats satisfaisant pour des références avoisinant  $0^\circ$  (sur le simulateur les références ne dépassaient pas les  $17^\circ$  pour les angles d'élévation et d'azimut).

## Annexe D

### Validation de la linéarisation

Afin de valider la linéarisation et de justifier le choix des points de fonctionnements, le calcul de l'erreur de linéarisation s'impose.

La formule de Taylor Lagrange pour le calcul du reste du développement de Taylor peut être une alternative à ce calcul celle-ci s'écrit comme suit :

Soit le développement de Taylor de la fonction  $f(x)$  au voisinage de  $a$ :

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \dots + \frac{f^{(n)}(a)}{n!}(x-a)^n + R(x)$$

Avec : (selon la formule d'Euler Lagrange :

$$R(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-a)^{n+1}$$

Avec  $\xi$  un nombre compris strictement entre  $a$  et  $x$ .

Dans le domaine de l'automatique, on fait appel au développement de Taylor dans le but de linéariser des systèmes. On s'arrête, donc, à l'ordre 1. La formule du reste devient alors :

$$R(x) = \frac{f^{(2)}(\xi)}{2}(x-a)^2$$

Puisque nous avons affaire à des fonctions vecteurs, le reste se calcul à l'aide du Hessien  $\mathcal{H}(f,x)$ .

Les voisinages des points de fonctionnement ont été choisis de manière à avoir une erreur entre le modèle linéaire et le modèle non linéaire inférieur à 10%.

De ce fait, suivant l'espace opérationnel, le nombre de point de fonctionnement a été fixé.