

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE

ECOLE NATIONALE POLYTECHNIQUE



Département du Génie Electrique
Spécialité Automatique

Projet de fin d'études

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

THÈME

**Synthèse et implémentation pratique de lois de
commandes non linéaires :
Application à un simulateur d'hélicoptère**

Etudié par :

**FARAH Faris
HACHID Faiçal**

Proposé et dirigé par :

**M^r. D. BOUKHETALA
M^r. M. TADJINE**

Juin 2005

Laboratoire de Commande des Processus
Ecole Nationale Polytechnique, 10, AV. Hassen Badi, El-Harrach, Algérie

Dédicaces

A qui puis je dédier cet humble travail si ce n'est à mes très chers parents, dont le sacrifice, la tendresse, l'amour, la patience, le soutien, l'aide et les encouragements sont l'essence de ma réussite. Sans eux, je ne serais pas ce que je suis aujourd'hui.

Je dédie ce travail également,

A tous les membres de la famille, ma soeur WIDAD, SOFIANE et AISSAM (pour sa délicieuse cuisine)

A 3amti ZAKIA et la famille ADEL, chesou, nabil et son fils walid, hamza, samira, naimaa ainsi que 3ami Abd'allah

A mes cousines et tantes, Saidaa, hadjiraa, farida et sa fille zineb (pour ses gâteaux)

A mon cousin LAREBI.

A Monsieur MOKRANI et monsieur AISSAOUI, mes professeurs de mathématique et physique au Lycée, à qui, je ne saurais comment exprimer ma profonde reconnaissance et gratitude pour ses perpétuels conseils et encouragements

A mes amis de polytech, guenifi, moh phoenix, massi et adel fawri

A mes amis automaticiens moh et son mratia, amiro (t'as de la chance), boussad (alors toi, sans commentaire) et faris (bon courage pour ta quête.)

A mes copains A.HAMACHE et R.MEBARKI, avec qui j'ai partagé des moments inoubliables de ma vie.

A tous mes amis de KHERRATA, yaciune, walid, walid, rida, wissam, simo.....

A emule (il se reconnaîtra)

Et à tous qui m'ont connu et aidé de près ou de loin dans la réalisation de ce travail.

Faïçal

DEDICACES

*Je d die ce travail   ma tr s ch re m re et   mon
p re, qui m'ont vraiment soutenu,   mes fr res et
s eurs, ainsi qu'  mes amis et camarades d' tudes.*

*A mon bin me fay al, « trin me » moh, Abd'nour
ainsi que amiro et boussad*

*Enfin que tous ceux qui ont particip  de pr s ou de
loin dans l' laboration de ce travail trouvent ici
l'expression*

de ma reconnaissance : ils se reconna tront

Paris

Remerciements



Le travail présenté dans ce mémoire a été mené au laboratoire de commande des processus du département de Génie Electrique de l'École Nationale Polytechnique.

Nous exprimons notre profonde gratitude à messieurs D. BOUKHETALA et M. TADJINE professeurs à l'E.N.P pour avoir assumé la lourde responsabilité de nous encadrer, nous orienter et de nous conseiller tout au long de la réalisation de ce travail ainsi pour la confiance qu'ils nous ont accordée.

Nos sincères et chaleureux remerciements à Monsieur F. BOUDJEMA pour son appui ainsi qu'aux conseils qu'il nous a fournis.

Nous tenons à remercier le président et les membres de jury pour nous avoir fait l'honneur d'évaluer notre travail.

Nous tenons également à remercier l'ensemble de nos enseignants d'automatique

Un grand merci à nos amis : A. HAMACHE et A. BOURZAK, pour leur aide et soutien dans les moments difficiles.

Enfin, nous remercions tout le personnel de la bibliothèque et tous ceux qui nous ont aidé de près ou de loin pour la réalisation de ce projet

TABLES DES MATIERES

INTRODUCTION GENERALE

I	PRESENTATION DU SIMULATEUR D'HELICOPTERE	3
I.1	INTRODUCTION	3
I.2	PRINCIPE DU VOL D'UN HELICOPTERE	3
I.2.1	Les forces agissantes sur l'hélicoptère en vol	3
I.2.2	Principe du vol de l'hélicoptère	4
I.3	DESCRIPTION DU SIMULATEUR	5
I.3.1	Propulseurs	6
I.3.2	Commande des moteurs	6
I.3.3	Les encodeurs optiques	7
I.3.4	La base du TRMS	8
I.3.5	Boîtier marche/arrêt	8
I.3.6	Le (SCSI BOX)	9
I.3.7	Carte d'acquisition : ADVANTECH PCI 1711	10
I.4	ENVIRONNEMENT DE DEVELOPPEMENT	11
I.5	CONCLUSION	15
II	MODELISATION DU SIMULATEUR	16
II.1	INTRODUCTION	16
II.2	MODELE NON LINEAIRE	17
II.2.1	Sous système d'élévation	17
II.2.1.1	Moment gravitationnel	18
II.2.1.2	Moment de la force aérodynamique	20

II.2.1.3	Moment des forces centrifuge	20
II.2.1.4	Moment de friction	21
II.2.1.5	Moment d'inertie	21
II.2.2	Sous système d'azimut	22
II.2.2.1	Moment de la force aérodynamique	22
II.2.2.2	Moment de friction	23
II.2.2.3	Moment d'inertie	23
II.2.3	Dynamiques des propulseurs (hélices + moteurs DC)	24
II.3	MODELE D'ETAT	25
II.4	LE MODELE DECOUPLE	27
II.4.1	Modèle 1 DDL vertical	27
II.4.2	Modèle 1DDL horizontal	27
II.5	PARAMETRES DU MODELE	28
II.6	SIMULATION	30
II.7	VALIDATION DU MODELE DE SIMULATION	31
II.7.1	Identification statique	31
II.8	CONCLUSION	36
III	RAPPELS THEORIQUES	37
III.1	INTRODUCTION	37
III.2	COMMANDE PAR LOGIQUE FLOUE	38
III.2.1	Notion de base sur la logique floue	38
III.2.1.1	Ensembles ordinaires	38
III.2.1.2	Ensemble flou	38
III.2.1.3	Variables linguistiques	39

III.2.1.4	Fonction d'appartenance	39
III.2.2	Raisonnement flou	40
III.2.3	Commande par la logique floue	41
III.2.3.1	Contrôleur flou	41
III.2.4	Types de régulateurs flous	42
III.2.4.1	Régulateur de type Mamdani	42
III.2.4.2	Régulateur de type Sugeno	43
III.2.5	Procédé lors de la conception d'un réglage par logique floue	43
III.3	COMMANDE PAR BOUCLAGE NON LINEAIRE	44
III.3.1	Théorie de la linéarisation des systèmes SISO non linéaires	44
III.3.1.1	Degré relatif	44
III.3.1.2	Forme normale	45
III.3.2	Linéarisation exacte par régulation d'état statique	48
III.3.2.1	Définition :	48
III.3.2.2	Linéarisation exacte d'un système en forme normale	48
III.3.2.3	Linéarisation exacte d'un système en forme quelconque	49
III.3.2.4	Sortie plate :	50
III.4	COMMANDE A STRUCTURE VARIABLE	51
III.4.1	Le mode glissant	52
III.4.1.1	Choix de la surface du glissement	52
III.4.1.2	Condition d'existence du régime glissant	54
III.4.1.3	Principe d'invariance	55
III.4.2	Le mode non glissant	56
III.4.2.1	Conditions d'attractivité	56
III.4.2.2	Approche directe	57

III.4.2.3	Approche de LYAPUNOV	57
III.4.3	Elimination du phénomène de chattering	58
III.5	CONCLUSION	60
IV	SYNTHESE DE LA COMMANDE	61
IV.1	INTRODUCTION	61
IV.2	COMMANDE FLOUE	62
IV.2.1	Structure de la commande floue	62
IV.2.1.1	Régulateur flou de MAMDANI	62
IV.2.1.2	Loi de commande	63
IV.2.1.3	Les gains de normalisation	63
IV.2.2	Synthèse des régulateurs	63
IV.2.3	L'action intégrale	64
IV.3	BOUCLAGE NON LINEAIRE	65
IV.3.1	Calcul du Degré relatif	65
IV.3.2	Synthèse de la loi de commande par bouclage non linéaire :	67
IV.3.3	Simulation	69
IV.4	COMMANDE PAR MODE DE GLISSEMENT	72
IV.4.1	Surface de glissement	72
IV.4.2	Calcul de la commande équivalente	73
IV.4.3	L'attractivité	74
IV.4.4	Etude de la dynamique réduite	75
IV.4.5	Simulation	76
IV.5	REGULATEUR FLOU GLISSANT	81
IV.5.1	Structure du contrôleur flou glissant	81

IV.5.2 Synthèse du contrôleur flou glissant	82
IV.5.2.1 Commande équivalente et surface de glissement	82
IV.5.2.2 Commande d'attractivité	82
IV.5.3 Simulation	82
IV.6 CONCLUSION	85
V IMPLEMENTATION ET RESULTATS EXPERIMENTAUX	86
V.1 INTRODUCTION	86
V.2 CONSIDERATIONS PRATIQUES	86
V.3 LA COMMANDE PAR LOGIQUE FLOUE	88
V.3.1 Stabilisation et poursuite	89
V.3.2 Variation paramétrique	93
V.3.3 Rejet de perturbation	95
V.4 COMMANDE PAR BOUCLAGE LINEARISANT	96
V.4.1 Réponse réelle VS réponse du modèle	97
V.4.2 Stabilisation et poursuite	98
V.4.3 Variation paramétrique	102
V.4.4 Rejet de perturbation	104
V.5 COMMANDE PAR MODE DE GLISSEMENT	105
V.5.1 Fonction d'attractivité : Sigmoide	106
V.5.1.1 Réponse réelle VS réponse du modèle	106
V.5.1.2 Stabilisation et poursuite	107
V.5.1.3 Variation paramétrique	111
V.5.1.4 Rejet de perturbation	113
V.5.2 Flou glissant	114

المكتبة الوطنية
BIBLIOTHEQUE
Ecole Nationale Polytechnique

V.5.2.1	Réponse réelle VS réponse du modèle	114
V.5.2.2	Stabilisation et poursuite	115
V.5.2.3	Variation paramétrique	119
V.5.2.4	Rejet de perturbation	121
V.6	ETUDE COMPARATIVE	122
V.7	CONCLUSION	123

CONCLUSION GENERALE

BIBLIOGRAPHIE

INTRODUCTION GENERALE



Pendant des décennies, les chercheurs de tous les domaines s'intéressaient aux systèmes aéronautiques. Plus particulièrement, les chercheurs automaticiens qui se sont penchés sur le problème de la commande des avions et des hélicoptères. En effet, ces engins sont des systèmes complexes non linéaires multivariables, instables dans certaines plages de vol et présentent une dynamique fortement couplée. Cependant, L'application des commandes développées en théorie n'est pas toujours systématique, des phases de validation dans des conditions réelles s'avèrent donc nécessaires.

Les avancées récentes en informatique et en électronique de puissance réduisant les coûts de fabrication des systèmes prototypes, ont rendu raisonnable le prix de la construction physique de simulateurs réels miniatures. Qui permettent de tester de nouvelles techniques de commandes. Comme le modèle mathématique n'est jamais précis et fidèle au phénomène réel, la commande robuste, est l'une des techniques investiguées, du fait qu'elle garantit la stabilité et les performances en présence d'erreurs de modélisations et de perturbations extérieures

L'objet du présent travail fait partie du même axe de recherche, qui consiste en la validation de différentes techniques de commandes non linéaires robustes sur un simulateur d'hélicoptère appelé TRMS (Twin Rotor MIMO System)

Ce manuscrit est structuré en cinq chapitres de la manière suivante

- Le premier chapitre est consacré à la présentation du système à commander, après avoir donné le principe du vol des hélicoptères, nous passons en revue les

différents constituants du simulateur ainsi que son principe de fonctionnement, nous nous focalisons ensuite sur son environnement de développement

- Dans le deuxième chapitre, nous procédons à la modélisation dynamique du simulateur, en premier lieu, un modèle couplé non linéaire sera développé, puis nous dérivons deux modèles à un degré de liberté qui seront utilisés ensuite pour la synthèse, ensuite nous essaierons de valider ces modèles et de les adapter au modèle réel, en procédant avec une identification statique des paramètres.
- Dans le troisième chapitre, le lecteur est initié à la théorie des différentes lois de commandes abordées. Ainsi, la commande par logique floue, la commande linéarisante et la commande par mode de glissement feront l'objet de ce chapitre
- Le quatrième chapitre, quant à lui, est consacré au développement et à la synthèse des lois de commandes qui seront appliquées sur le modèle du simulateur, la stratégie de commande adoptée est la commande décentralisée, on terminera ce chapitre, par la proposition d'une commande hybride, à savoir « la commande floue glissante », qui sert à éliminer un inconvénient de la commande par mode de glissement
- Le chapitre V, fera le lien entre les chapitres, en exposant les résultats expérimentaux de l'implémentation temps réel des lois de commandes synthétisées dans le chapitre IV, sur le simulateur d'hélicoptère présenté et modélisé dans les chapitres I et II. Sans oublier de donner quelques considérations pratiques.

Le travail s'achèvera par une conclusion générale avec des perspectives pour continuer sur le même objectif.

CHAPITRE I

***PRESENTATION DU SIMULATEUR
D'HELICOPTERE***



I PRÉSENTATION DU SIMULATEUR D'HELICOPTERE

I.1 INTRODUCTION

Les systèmes aéronautiques sont devenus aujourd'hui si difficiles et complexes qu'on ne puisse pas réaliser leurs commandes par des techniques classiques. En effet, des chercheurs automaticiens se sont penchés sur ces problèmes de commande, pour cela plusieurs prototypes ont été réalisés dans le but de tester de nouvelles techniques de commande. Le simulateur d'hélicoptère (TRMS) est l'un de ces prototypes sur lequel nous allons travailler.

Dans ce chapitre, nous allons d'abord donner un petit aperçu sur le principe du vol des hélicoptères, ensuite nous présentons en détail le simulateur.

I.2 PRINCIPE DU VOL D'UN HELICOPTERE [SITE] [SAMB91][KAFI99]

I.2.1 Les forces agissantes sur l'hélicoptère en vol

Nous supposons, pour simplifier la présentation du vol de l'hélicoptère, que les pales du rotor principal décrivent un disque dont l'axe passe par le centre de gravité de l'hélicoptère.

En vol, l'hélicoptère est soumis à trois forces [figure I.1] :

- Le poids, appliqué au centre de gravité.
- La force générée par le rotor (portance).
- La traînée générale de l'appareil provoquée en vol de translation, par résistance de l'air sur la structure.

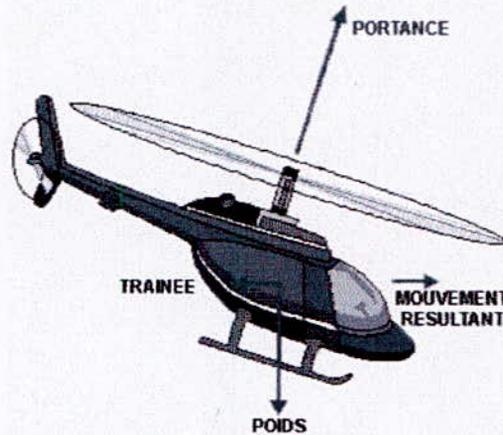


Figure I.1 forces agissantes sur l'hélicoptère

Pour que l'hélicoptère soit en équilibre, il faut que la résultante du poids et de la traînée soit égale et opposée à la force générée par le rotor. La résultante des forces est alors nulle et le vol est stabilisée : si l'hélicoptère est immobile il le demeure, s'il est soumis à une translation, son mouvement est rectiligne. Le mouvement de l'hélicoptère résulte des variations en intensité et en direction de la portance.

1.2.2 Principe du vol de l'hélicoptère

Le principe du vol est à la fois simple et très compliqué, tout dépend du niveau de sophistication de l'explication. Regardons dans les grandes directions le principe du vol de l'hélicoptère est le suivant :

L'hélice de l'avion assure une traction, elle tire l'avion en se vissant dans l'air comme une vis dans le bois, lorsqu'au décollage l'avion va assez vite son aile génère une portance qui lui permet de voler. [Figure I.2]

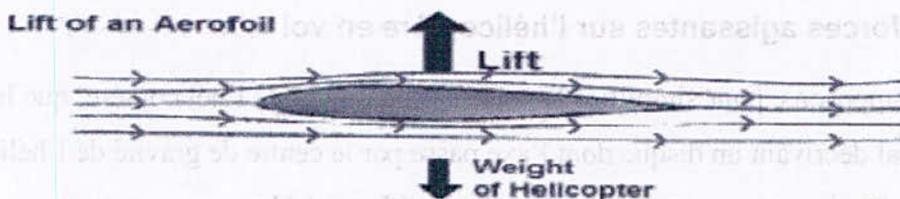


Figure I.2 principe du vol de l'hélicoptère

Le rotor de l'hélicoptère est à la fois l'aile et l'hélice, il assure traction et portance en tournant d'où son appellation rotor.

I.3 DESCRIPTION DU SIMULATEUR [MAN1]

Le TRMS (Twin Rotor Mimo System) est un système physique aérodynamique conçu pour le développement et l'implémentation de nouvelles lois de commandes. Ce-ci inclut, la modélisation de la dynamique du système, l'identification, l'analyse et la conception de divers contrôleurs par des méthodes classiques et modernes. Son comportement ressemble à celui d'un hélicoptère, de point de vue de commande ; il présente un système non linéaire d'ordre supérieur avec des couplages significatifs.

Le TRMS est formé de [figure I.3]:

- une poutre qui peut pivoter sur sa base de telle manière qu'elle puisse tourner librement dans le plan vertical et horizontal.
- Deux propulseurs (principal et secondaire) fixées aux deux extrémités de la poutre, ils sont formés d'une hélice, un moteur à courant continu ainsi qu'un bouclier pour des raisons de sécurité
- Un contrepoids fixé sur la tige à son pivot, son rôle est de diminuer les vibrations (oscillations) de la poutre
- Une tour pour maintenir la poutre
- Une base comprenant des circuits électroniques pour l'adaptation, synchronisation et filtrage des signaux entrants et sortants
- Un boîtier de marche/arrêt des moteurs

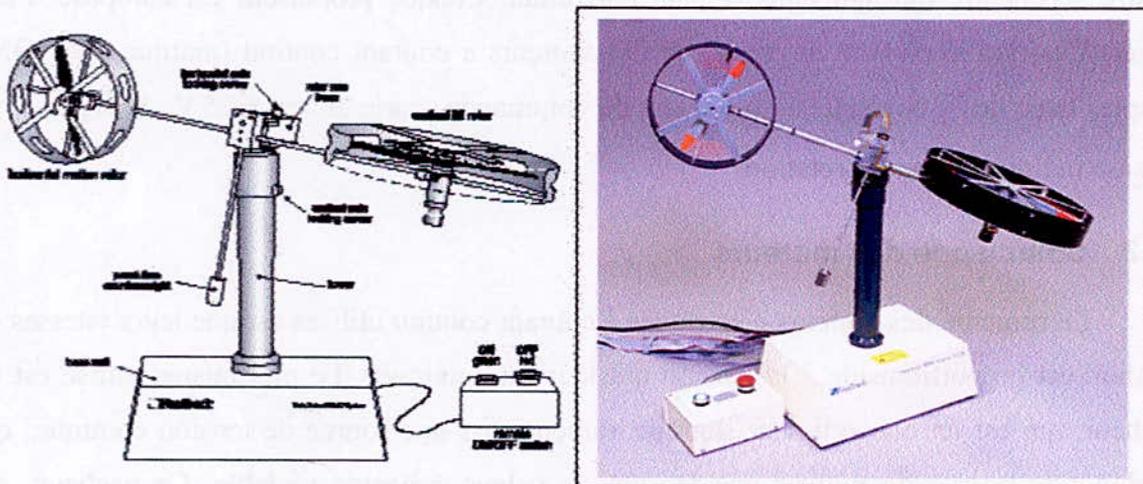


Figure I.3 : le simulateur d'hélicoptère TRMS
(Twin Rotor Mimo System)

Le modèle du TRMS est multi-variable à deux entrées et deux sorties (MIMO 2*2). Les sorties étant l'angle d'élévation et l'angle d'azimut, les entrées étant les tensions appliquées aux deux moteurs. 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 par de 2 vis. [figure I.4]

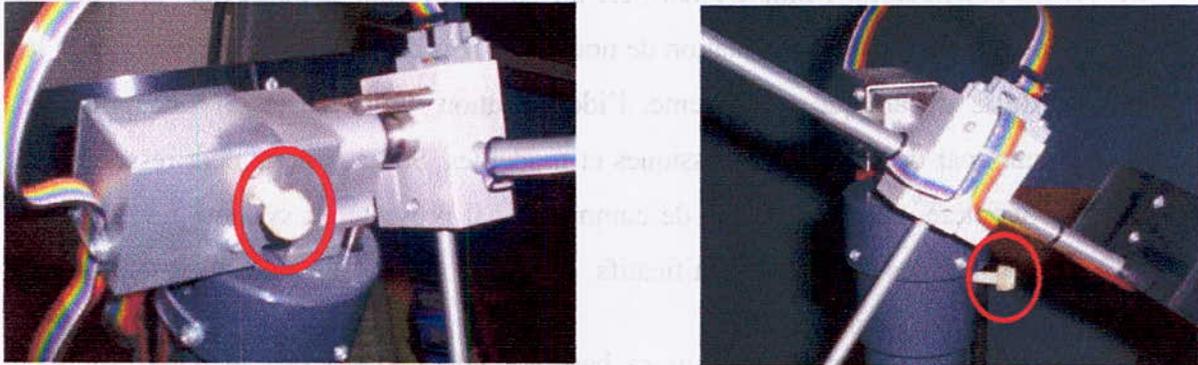


Figure I.4 : les deux vis de fixation

L'objectif de commande est de faire pivoter la poutre autour des deux axes rapidement et avec précision, et cela en variant les forces aérodynamiques des deux propulseurs. Dans un hélicoptère normal la force aérodynamique est commandée en changeant l'angle d'attaque. Néanmoins dans le TRMS cet angle est fixé, et les forces aérodynamiques sont commandées en changeant la vitesse des propulseurs.

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

Le principe des moteurs électriques à courant continu utilisés est que leurs vitesses de rotation est proportionnelle 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 qui produit 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.

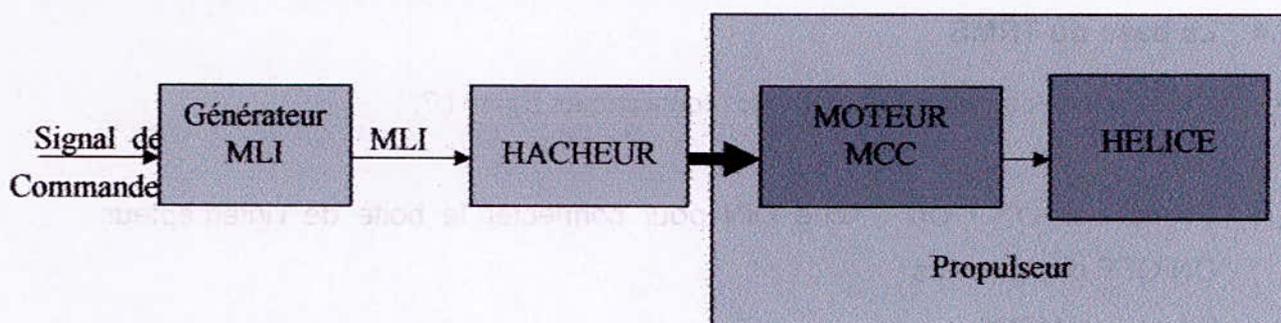


Figure I.5 : schéma synoptique du fonctionnement du propulseur

I.3.3 Les encodeurs optiques

Le TRMS a deux capteurs pour mesurer l'orientation de la poutre dans l'espace, ils se situent à l'intérieur du pivot sphérique de la poutre. Il s'agit de deux encodeurs optiques incrémentaux : le principe de fonctionnement est le suivant [figure I.6]:

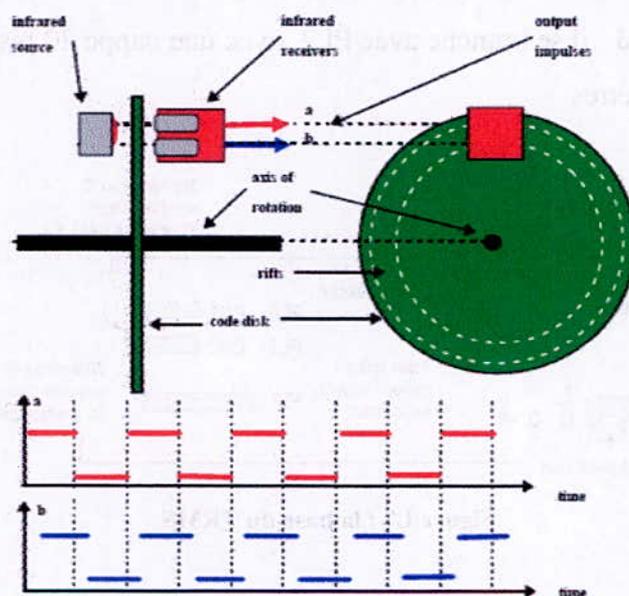


Figure I.6 : encodeurs optiques

Les faisceaux lumineux émis par les deux photo diodes (A et B) passent par deux anneaux de fentes sur le disque. Les fentes ont une différence de phase, de sorte que la tension des récepteurs (A et B) soient des ondes rectangulaires avec une différence de phase. Le signe de la différence de phase permet de déterminer la direction de la rotation

I.3.4 La base du TRMS

À l'arrière de la base on trouve les différentes connectiques figure I.7:

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

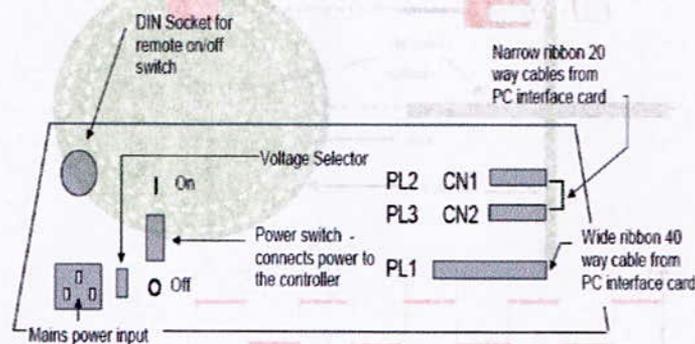


Figure I.7 : la base du TRMS

I.3.5 Boîtier marche/arrêt

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



I.3.6 Le (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 aussi 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 commandes des moteurs)
- un autre petit connecteur PL3 pour les entrées analogiques (retour tachymétrique des deux moteurs)

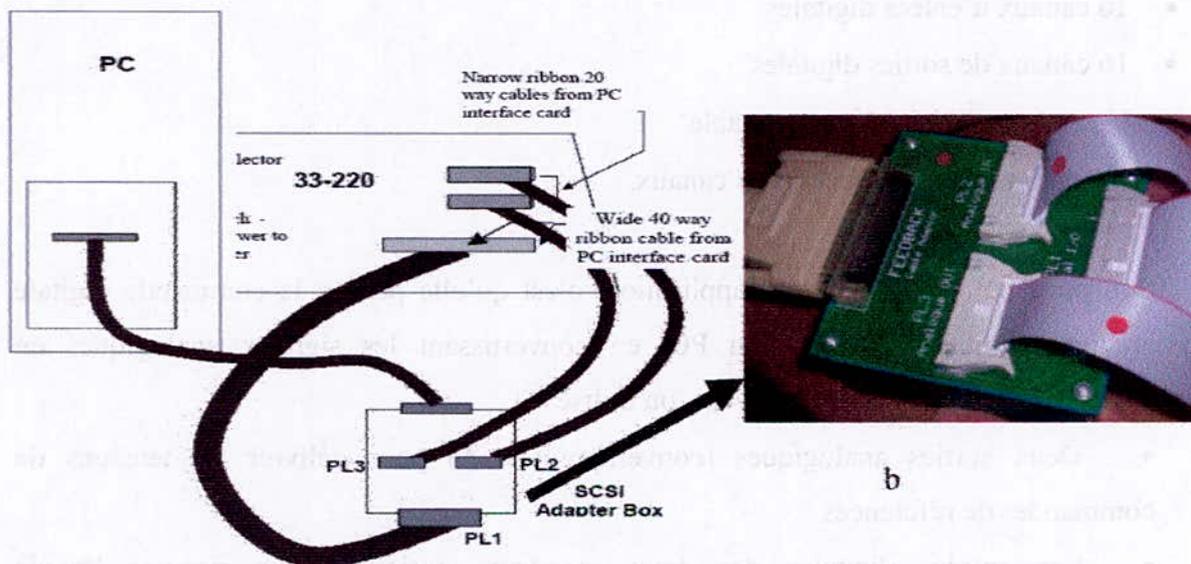


Figure I.8 : branchement des différents connecteurs

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 digitaux. 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 entées analogiques est programmables et contrôlable par software *
- Chaque canal à sa gamme individuelle stockée dans la RAM de la carte
- 2 sorties analogiques (convertisseur D/A)
- 16 canaux d'entées digitales
- 16 canaux de sorties digitales
- Un compteur/timer programmable
- Scanne automatique des gains/canaux.

Son principal rôle dans notre application, c'est qu'elle permet 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 commandes 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 DEVELOPPEMENT [RTW][RTWINTGT][MANI]

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 cause des pertes de temps considérables, en testant et en calibrant à nouveau le régulateur implémenté, voici ci-dessous, un petit comparatif entre cette approche et l'approche classique :

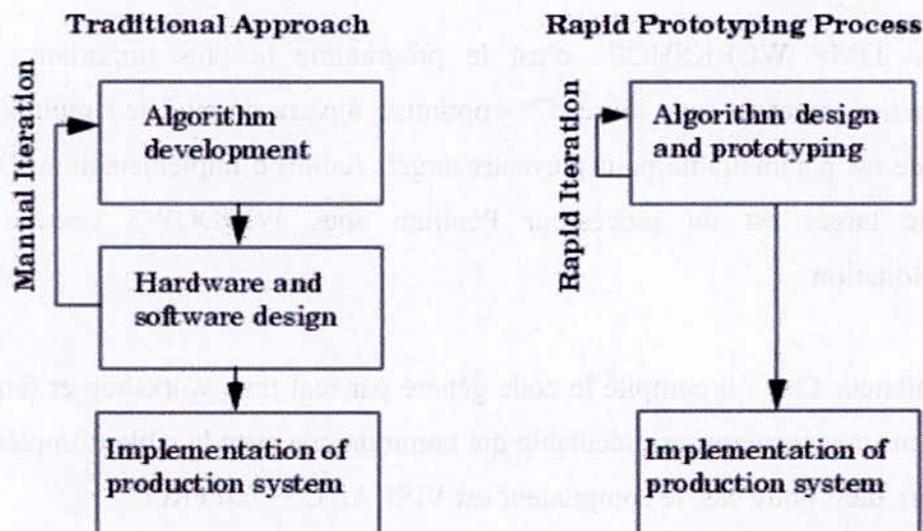


Figure I.9 : rapid prototyping VS processus de développement traditionnel

Avec le « hardware-in-the-loop », on teste directement le contrôleur qu'on a synthétisé en simulation sur le système physique réel, avec un simple PC de développement. En effet tous ce qu'on a besoin est :

- 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 nécessaires

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, avec lequel on programme avec des objets ou blocks, ces objets peuvent être soit des block standards de bien des blocks spécialisées écrit par l'utilisateur comme des S-fonctions
- **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), dans notre cas, le compilateur est 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 donc il n'est pas 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 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 aussi, des bibliothèques, ou drivers de la carte d'acquisition, pour les intégrer dans notre programme

Voici ci-dessus un schéma [figure I.10] qui illustre la relation qui existe entre les différents modules :

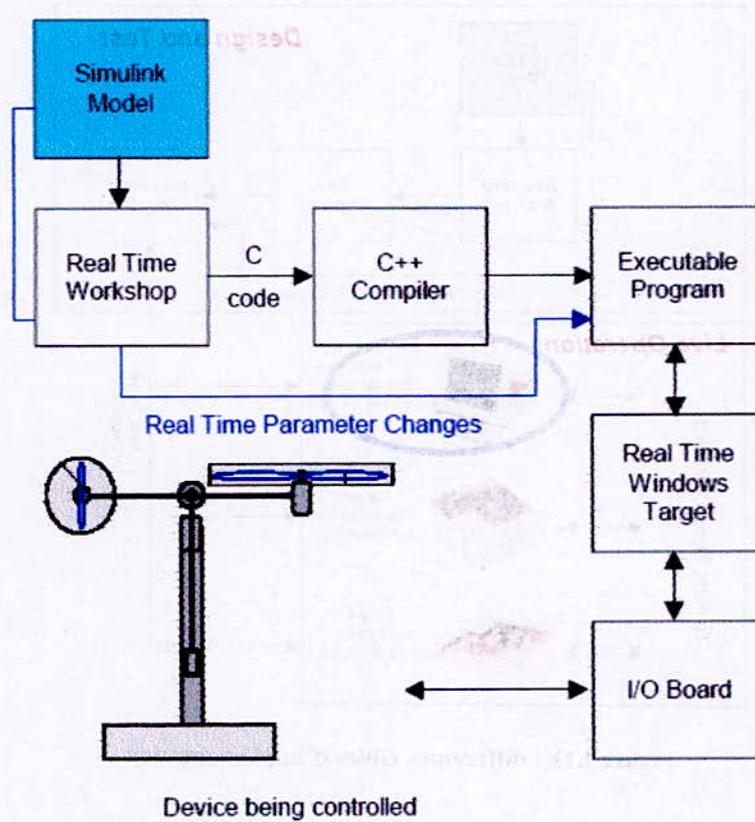


Figure I.10 : processus de génération du code exécutable



Et voici un autre schéma [figure I.11] qui illustre les différentes cibles (targets) et les différentes manières dont on peut implémenter notre régulateur

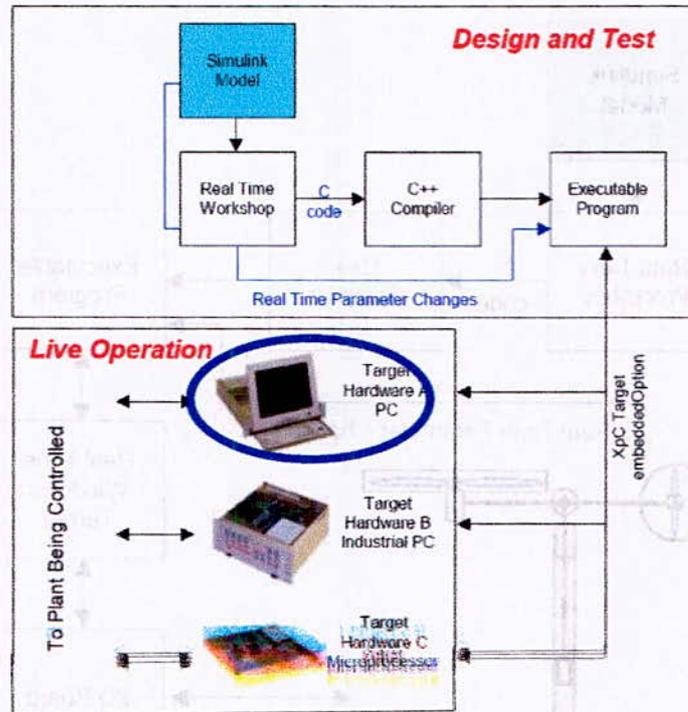


Figure I.11 : différentes cibles d'implémentation

Pour passer d'un régulateur simulé (non temps réel) vers un programme temps réel, il suffit juste 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 voici la librairie utilisée :

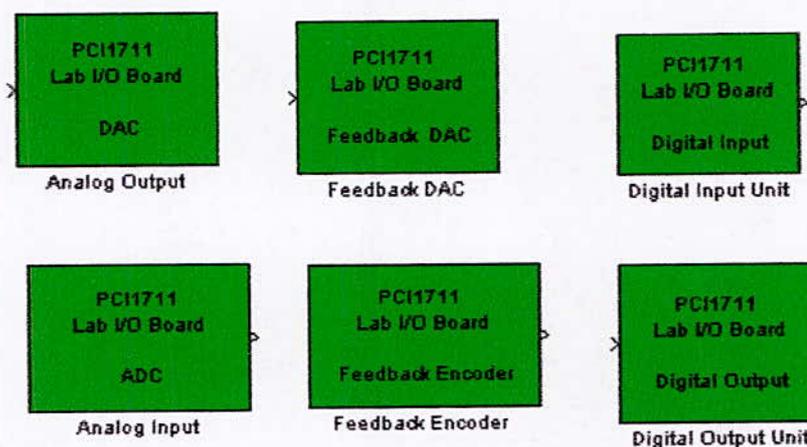


Figure I.12 : librairie de la carte d'acquisition

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

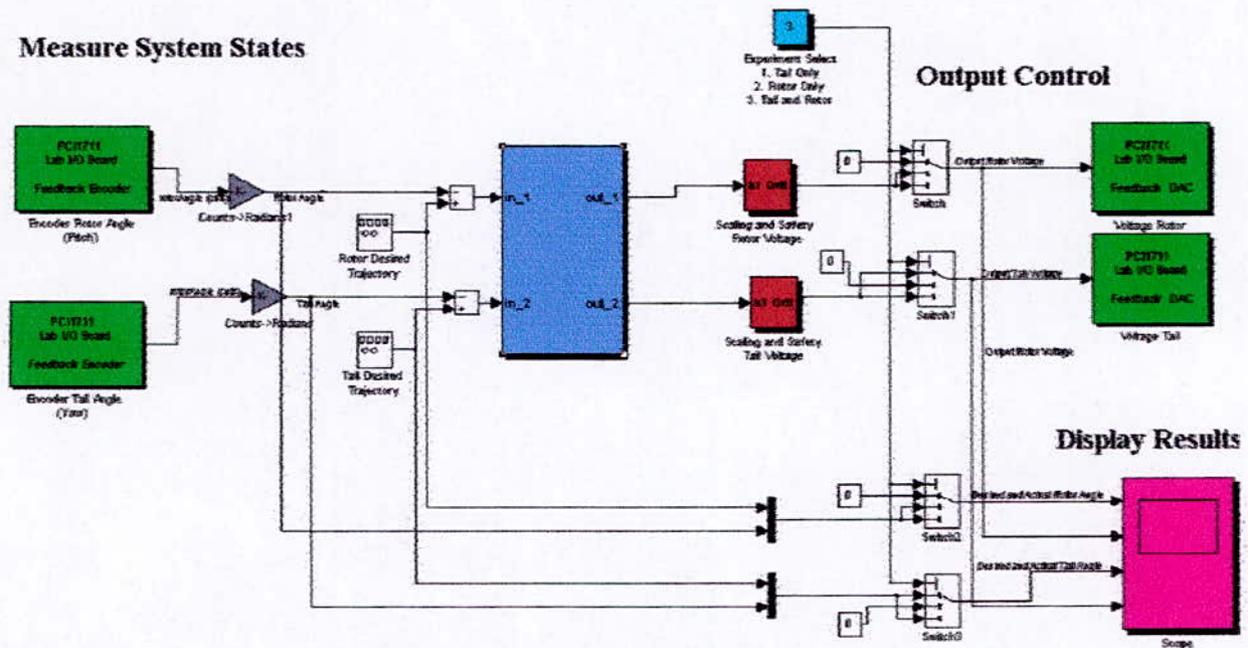


Figure I.13 : exemple d'implémentation d'un contrôleur

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 $\pm 2.5\text{ V}$

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 par 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 juste 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 présenté une brève description du simulateur (TRMS). Ainsi que son principe de fonctionnement. Enfin, nous avons expliqué l'environnement de développement utilisé.

CHAPITRE II

MODELISATION DU SIMULATEUR

II MODELISATION DU SIMULATEUR

II.1 INTRODUCTION

Pour effectuer l'analyse et la synthèse dynamique, il est nécessaire de connaître les relations entre les grandeurs d'entrées et de sorties. L'ensemble de ces relations constitue le modèle mathématique du système. L'efficacité de ce modèle repose sur une analogie entre le comportement des objets physiques et celui des formules mathématiques.

Dans l'automatique, modéliser un système consiste à établir un ensemble de relations mathématiques qui permettent de décrire avec précision suffisante les interactions entre le système et son environnement. Lorsque les relations citées auparavant sont issues des équations de la physique, le modèle obtenu est dit modèle de connaissance, ces relations découlent des observations disponibles sur le système. On aboutit aussi au modèle de représentation, en passant par l'identification.

Dans le présent chapitre, nous abordons en premier la modélisation dynamique du système déjà présenté dans le chapitre I. Ensuite, nous validerons le modèle développé, et en fin nous présenterons des simulations en boucle ouverte du modèle.

II.2 MODELE NON LINEAIRE [MAN2]

Le modèle physique est développé sous certaines hypothèses simplificatrices. En premier lieu, on suppose que les dynamiques du sous-système rotor sont des équations différentielles du premier ordre. De plus, on suppose que les frottements sont de type visqueux, et que le sous-système hélice-air peut être décrit par les lois d'écoulement aérodynamiques.

II.2.1 Sous système d'élévation

D'abord, considérons la rotation de la poutre dans le plan vertical. C'est-à-dire autour de l'axe horizontal. En appliquant la seconde loi de Newton on obtient :

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

Avec :

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

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

L'équation (II.1) peut être écrite sous la forme :

$$J_v \frac{d^2 \alpha_v}{dt^2} = M_{v1} + M_{v2} + M_{v3} + M_{v4} \quad (II.4)$$

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
- α_v : L'angle d'élévation de la tige
- 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

Les différents moments sont calculés comme suit :

II.2.1.1 Moment gravitationnel M_{vi}

Pour déterminer les moments de la gravitation appliqués à la poutre et qui la mettent en rotation autour de l'axe horizontal, on considère la situation illustrée dans la figure [II.1]

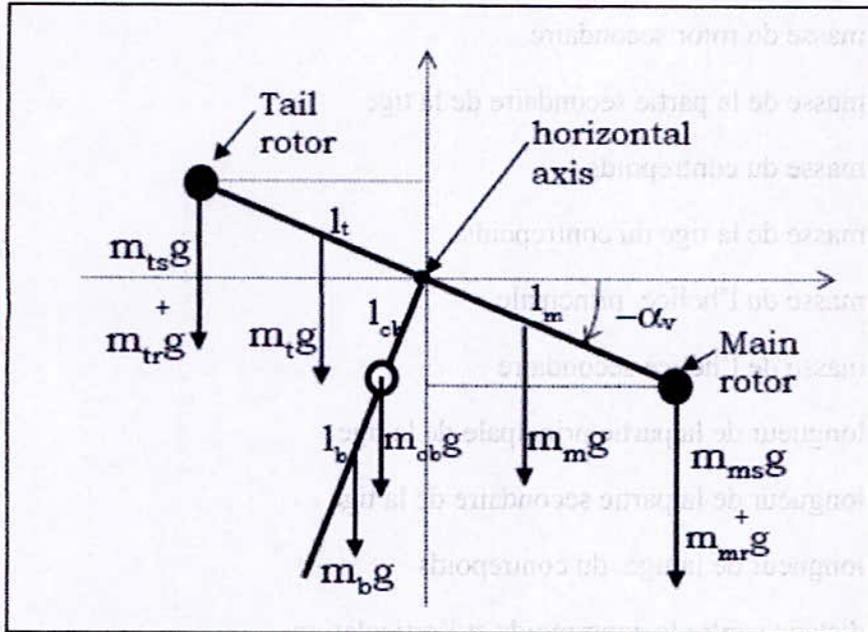


Figure II.1: forces de gravités agissantes sur le TRMS

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

Qui peut être écrite :

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

$$\text{Avec : } \begin{cases} A = \left(\frac{m_t}{2} + m_{tr} + m_{ts} \right) l_t \\ B = \left(\frac{m_m}{2} + m_{mr} + m_{ms} \right) l_m \\ C = \left(\frac{m_b}{2} l_b + m_{cb} l_{cb} \right) \end{cases} \quad (\text{II.7})$$

Où :

M_{v_1} : Le moment correspondant aux forces de gravités

m_{mv} : La masse du rotor principal

m_m : La masse de la partie principale de la tige

m_{tr} : La masse du rotor secondaire

m_t : La masse de la partie secondaire de la tige

m_{cb} : La masse du contrepoids

m_b : La masse de la tige 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 tige

l_t : La longueur de la partie secondaire de la tige

l_b : La longueur de la tige du contrepoids

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

g : L'accélération gravitationnelle

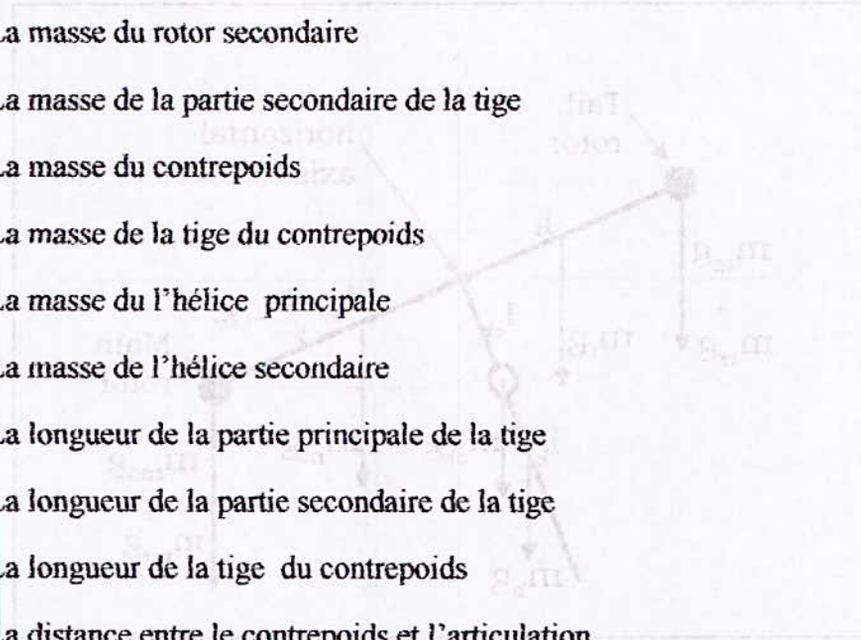


Figure II.1 : forces de gravité

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

Pour déterminer les moments des forces propulsives appliquées à la tige on considère la situation suivante illustrée dans la figure [II.2]

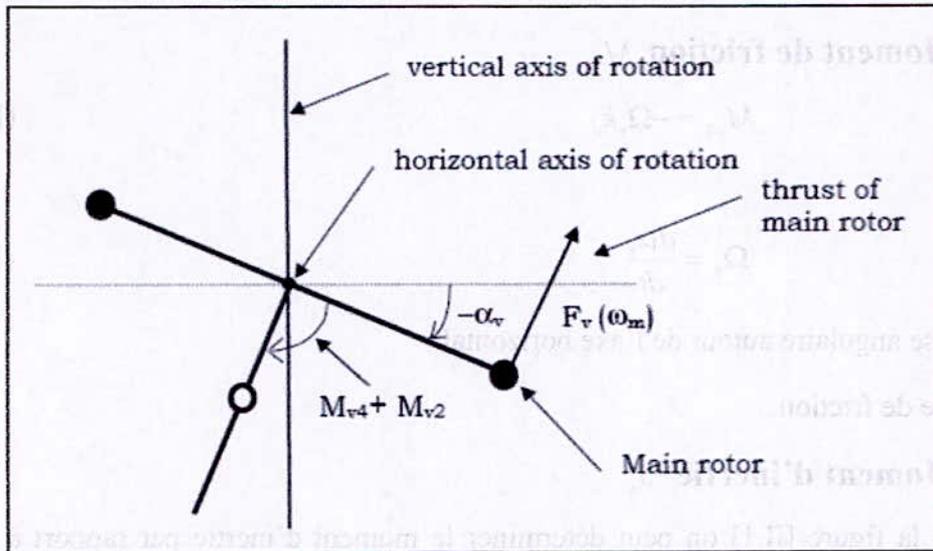


Figure II.2: Moments de la force aérodynamique et de friction

$$M_{v2} = l_m F_v(\omega_m) \quad (\text{II.8})$$

Où :

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

ω_m : La vitesse angulaire du rotor principal

$F_v(\omega_m)$: Exprime la dépendance de la force aérodynamique de la vitesse angulaire du moteur. Elle doit être mesurée expérimentalement

II.2.1.3 Moment des forces centrifuge M_{v3}

$$M_{v3} = -\Omega_h^2 \left\{ \left(\frac{m_t}{2} + m_{tr} + m_{ts} \right) l_t + \left(\frac{m_m}{2} + m_{mr} + m_{ms} \right) l_m + \left(\frac{m_b}{2} l_b + m_{cb} l_{cb} \right) \right\} \sin \alpha_v \cos \alpha_v \quad (\text{II.9})$$

Avec :

$$\Omega_h = \frac{d\alpha_h}{dt} \quad (\text{II.10})$$

Ω_h : Vitesse angulaire de la tige autour de l'axe vertical et α_h est l'angle d'azimut de la tige

On peut écrire (II.9) sous forme compacte :

$$M_{v3} = -\Omega_h^2 (A+B+C) \sin \alpha_v \cos \alpha_v \quad (\text{II.11})$$

II.2.1.4 Moment de friction M_{v4}

$$M_{v4} = -\Omega_v k_v \quad (\text{II.12})$$

Avec :

$$\Omega_v = \frac{d\alpha_v}{dt} \quad (\text{II.13})$$

Ω_v : La vitesse angulaire autour de l'axe horizontal

k_v : Constante de friction.

II.2.1.5 Moment d'inertie J_v

D'après la figure [II.1] on peut déterminer le moment d'inertie par rapport à l'axe horizontal. On remarque que ce moment est indépendant de l'angle d'élévation

$$\left\{ \begin{array}{l} J_{v1} = m_m l_m^2 \\ J_{v2} = m_m \frac{l_m^2}{3} \\ J_{v3} = m_{cb} l_{cb}^2 \\ J_{v4} = m_b \frac{l_b^2}{3} \\ J_{v5} = m_t l_t^2 \\ J_{v6} = m_t \frac{l_t^2}{3} \\ J_{v7} = \frac{m_{ms}}{2} r_{ms}^2 + m_{ms} l_m^2 \\ J_{v8} = m_{ts} r_{ts}^2 + m_{ts} l_t^2 \end{array} \right. \quad (\text{II.14})$$

ou :

r_{ms} Le rayon de l'hélice principale

r_{ts} Le rayon de l'hélice secondaire

II.2.2 Sous système d'azimut

De la même façon, on peut décrire le mouvement de la tige autour de l'axe vertical, le mouvement horizontal peut être décrit comme étant un mouvement de rotation d'un solide :

$$M_h = J_h \frac{d^2 \alpha_h}{dt^2} \quad (\text{II.15})$$

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 :

$$M_h = \sum_{i=1}^2 M_{hi} \quad (\text{II.16})$$

$$J_h = \sum_{i=1}^8 J_{hi} \quad (\text{II.17})$$

II.2.2.1 Moment de la force aérodynamique

Pour déterminer les moments de forces appliquées à la tige, on considère le cas présenté dans la figure [II.3]

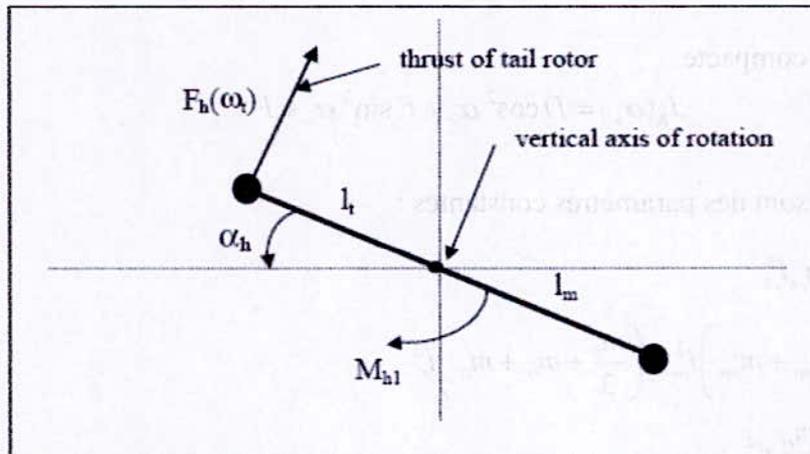


Figure II.3 : moments des forces dans le plan horizontal

$$M_{hi} = l_i F_h(\omega_t) \cos \alpha_v \quad (\text{II.18})$$

Où :

- ω_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.2.2.2 Moment de friction

$$M_{h2} = -\Omega_h k_h \quad (\text{II.19})$$

Avec : k_h Constante de friction

II.2.2.3 Moment d'inertie

Les moments d'inertie relative à l'axe vertical sont :

$$\left\{ \begin{array}{l} J_{h1} = \frac{m_m}{3} (l_m \cos \alpha_v)^2 \\ J_{h2} = \frac{m_l}{3} (l_l \cos \alpha_v)^2 \\ J_{h3} = \frac{m_b}{3} (l_b \sin \alpha_v)^2 \\ J_{h4} = m_{lr} (l_l \cos \alpha_v)^2 \\ J_{h5} = m_{mr} (l_m \cos \alpha_v)^2 \\ J_{h6} = m_{cb} (l_{cb} \sin \alpha_v)^2 \\ J_{h7} = \frac{m_{ls}}{2} r_{ls}^2 + m_{ls} (l_{ll} \cos \alpha_v)^2 \\ J_{h8} = m_{ms} r_{ms}^2 + m_{ms} (l_m \cos \alpha_v)^2 \end{array} \right. \quad (\text{II.20})$$

Où sous forme compacte :

$$J_h(\alpha_v) = D \cos^2 \alpha_v + E \sin^2 \alpha_v + F \quad (\text{II.21})$$

Où : D, E et F sont des paramètres constantes :

$$\left\{ \begin{array}{l} D = \frac{m_b}{3} l_b^2 + m_{cb} l_{cb}^2 \\ E = \left(\frac{m_m}{3} + m_{mr} + m_{ms} \right) l_m^2 + \left(\frac{m_l}{3} + m_{lr} + m_{ls} \right) l_l^2 \\ F = m_{ms} r_{ms}^2 + \frac{m_{ls}}{2} r_{ls}^2 \end{array} \right. \quad (\text{II.22})$$

Où :

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

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

S_v Le moment angulaire dans le plan vertical.

S_h Le moment angulaire dans le plan horizontal.

II.2.3 Dynamiques des propulseurs (hélices + moteurs DC) [bald02]

Les deux moteurs sont à courant continu commandés en tension. On considère le modèle simple d'une MCC avec une charge extérieure :

$$I \dot{\omega} = \frac{K_t}{R} (u - K_b \omega) - T_L \quad (\text{II.23})$$

Avec :

ω : 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_t : Constante du couple.

T_L : Couple résistant généré par la charge.

La charge, T_L , ici représente les frottements mécaniques et les frottements aérodynamiques générés par la rotation de l'hélice. Puisque ces frottements sont difficiles à modéliser, on a introduit une nouvelle variable U_{vv} pour le vertical et U_{hh} pour l'horizontal ainsi que deux fonctions non linéaires statiques. Ensuite, il suffit de déterminer leurs caractéristiques statiques expérimentalement

Le sous-système devient alors un système de premier ordre avec une fonction non linéaire à sa sortie :

$$\begin{cases} \frac{du_{vv}}{dt} = \frac{1}{T_{mv}} (-u_{vv} + K_{mv} u_v) \\ \omega_m = p_v(u_{vv}) \end{cases} \quad (\text{II.24})$$

$$\begin{cases} \frac{du_{hh}}{dt} = \frac{1}{T_{mr}} (-u_{hh} + K_{mr} u_h) \\ \omega_t = p_h(u_{hh}) \end{cases} \quad (\text{II.25})$$

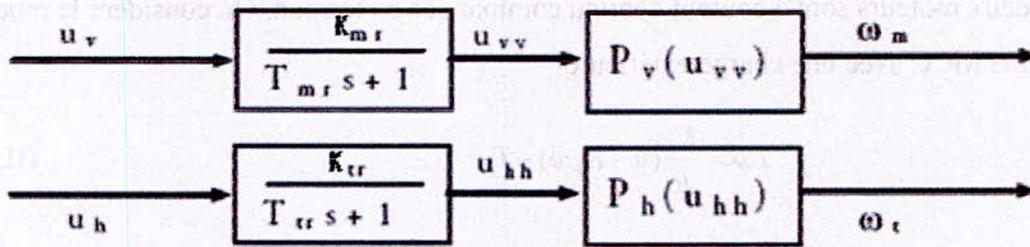


Figure II.4 : Schéma bloc des moteurs

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 .

II.3 MODELE D'ETAT

En utilisant les équations précédentes, on peut écrire les équations décrivant le mouvement du système comme suit :

$$\frac{dS_v}{dt} = \frac{I_m F_v(\omega_m) - \Omega_v k_v + g((A-B)\cos\alpha_v - c\sin\alpha_v) - \frac{1}{2}\Omega_h^2(A+B+C)\sin 2\alpha_v}{J_v} \quad (\text{II.26})$$

$$\frac{d\alpha_v}{dt} = \Omega_v \quad (\text{II.27})$$

$$\Omega_v = \frac{S_v + J_{tr}\omega_t}{J_v} \quad (\text{II.28})$$

$$\frac{dS_h}{dt} = \frac{I_t F_h(\omega_t)\cos\alpha_v - \Omega_h k_h}{J_h(\alpha_v)} \quad (\text{II.29})$$

$$\frac{d\alpha_h}{dt} = \Omega_h \quad (\text{II.30})$$

$$\Omega_h = \frac{S_h + J_{mr}\omega_m \cos\alpha_v}{J_h(\alpha_v)} \quad (\text{II.31})$$

En choisissant comme :

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

On obtient le modèle d'état ci-dessous :

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

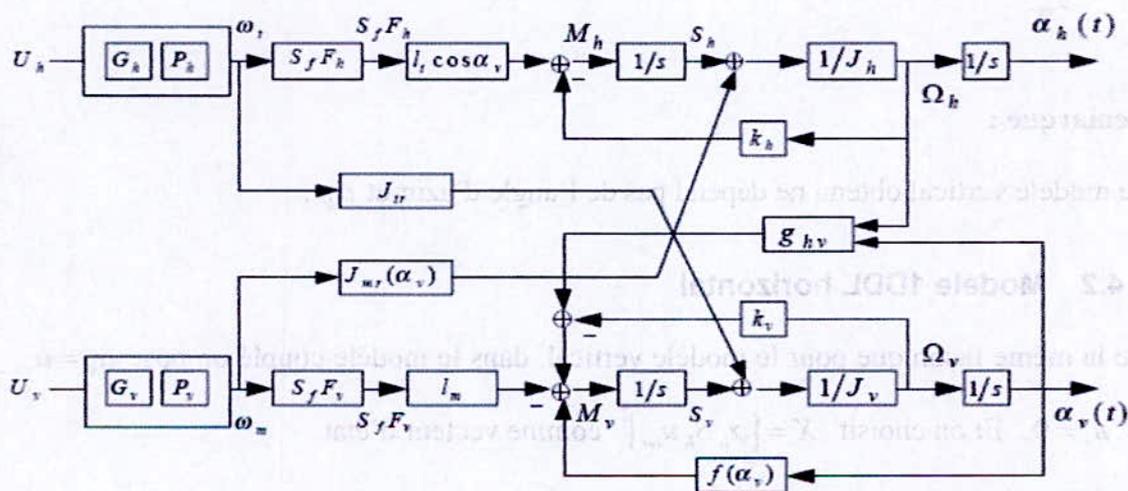


Figure II.5 : Schéma bloc du TRMS

II.4 LE MODELE DECOUPLE

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

II.4.1 Modèle 1 DDL vertical

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

On choisit le vecteur d'état suivant :

$$X = [\alpha_v, S_v u_v]^T \tag{II.33}$$

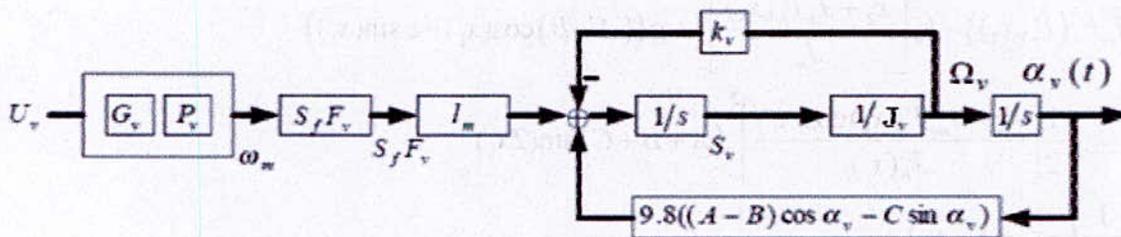


Figure II.5 : Schéma bloc du modèle vertical

La représentation d'état est alors :

$$\begin{cases} \dot{x}_1 = \frac{1}{J_v} x_2 \\ \dot{x}_2 = l_m F_v(P_v(x_3)) - k_v x_2 + g((A-B)\cos(x_1) - c\sin(x_1)) \\ \dot{x}_3 = \frac{1}{I_{mr}} (-x_3 + K_{mr} u_v) \end{cases} \tag{II.34}$$

Remarque :

Le modèle vertical obtenu ne dépend pas de l'angle d'azimut α_h .

II.4.2 Modèle 1DDL 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 $X = [\alpha_h, S_h u_h]^T$ comme vecteur d'état.

Le modèle horizontal est ainsi :

$$\begin{cases} \dot{x}_1 = \frac{1}{J_h(\alpha_{v0})} x_2 \\ \dot{x}_2 = l_r F_h(P_h(x_3)) \cos(\alpha_{v0}) - k_h x_2 \\ \dot{x}_3 = \frac{1}{T_{tr}} (-x_3 + K_{tr} u_h) \end{cases} \quad (II.35)$$

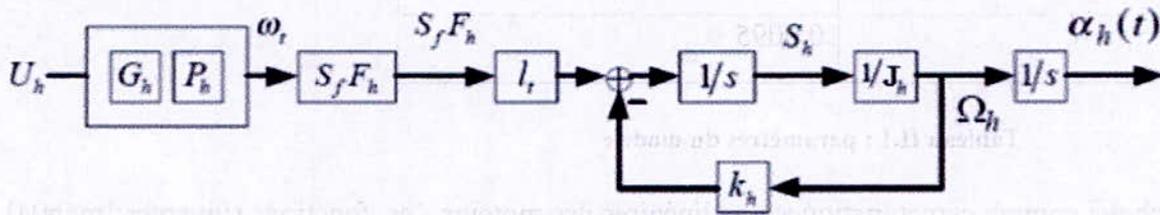


Figure II.6 Schéma bloc du modèle horizontal

II.5 PARAMETRES DU MODELE

Pour simuler les modèles précédents, on doit d'abord déterminer les paramètres du modèle, qu'on peut les diviser en trois catégories :

- Paramètres physiques.
- Caractéristiques non linéaires.
- Les constantes de temps et les gains statiques.

Paramètre	Valeur
A	0.0947
B	0.1105
C	0.011702
D	0.048814
E	0.0016087
F	0.006225
J_v	0.055448
J_{mr}	$1.6543 \cdot 10^{-5}$
J_{tr}	$2.65 \cdot 10$
l_m	0.24

$l_r(m)$	0.25
T_{mr}	1.432
T_{tr}	0.3842
K_{mr}	1
K_{tr}	1
K_v	0.00545371
K_h	0.0095

Tableau II.1 : paramètres du modèle

On a choisi comme caractéristiques non linéaires des moteurs, les fonctions suivantes:[man04]

$$\begin{cases} P_v(u_w) = \text{atan}(3u_w) \\ P_h(u_{th}) = \text{atan}(u_{th}) \end{cases} \quad (II.36)$$

Et comme fonctions aérodynamiques les fonctions suivantes :[man04]

$$\begin{cases} F_v(\omega_m) = 0.072183\omega_m^3 - 0.0027708\omega_m^2 + 0.019151\omega_m \\ F_h(\omega_t) = 0.07254\omega_t^3 - 0.075899\omega_t^2 + 0.037587\omega_t \end{cases} \quad (II.37)$$

Paramètre	Valeur
a	0.0047
b	0.102
c	0.01702
d	0.042814
e	0.0010083
f	0.006322
g	0.022428
h	1.842.10 ⁻⁴
i	1.62.10 ⁻⁴
j	0.24

II.6 SIMULATION

Voici ci-dessous quelques réponses du système en boucle ouverte

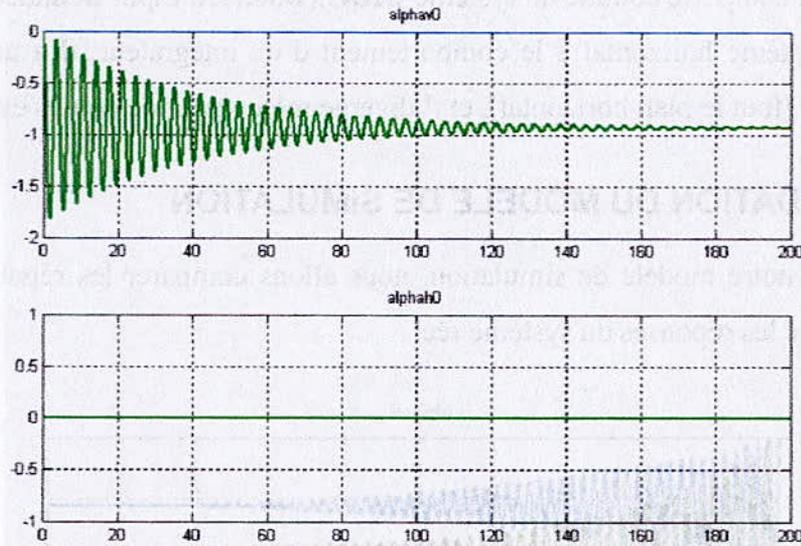


Figure II.7 Réponse libre du système avec $\alpha_{v0} = 0$ et $\alpha_{h0} = 0$ et $u_v = u_h = 0$

Bleu : référence vert : sortie du modèle

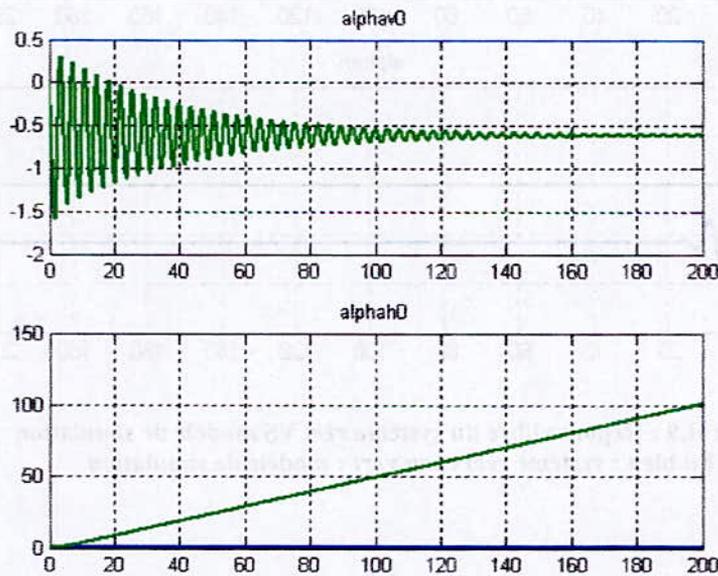


Figure II.8 : Réponse du système pour $u_v = u_h = 0.5$ avec $\alpha_{v0} = 0$ et $\alpha_{h0} = 0$

Bleu : référence vert : sortie du modèle

La réponse libre du sous-système d'élévation est oscillatoire amortie, dont l'origine est le moment gravitationnel, ce dernier n'agit que dans le plan vertical. L'amortissement quant à lui est dû aux forces de frictions qui forcent le système à se stabiliser à un point d'équilibre ($\alpha_{v0} = -0.93 \text{ rad}$). Quant au sous système d'azimut, il reste à sa position initiale tant qu'il n'est pas excité.

Lorsqu'on excite le système, le sous-système vertical tend vers un nouveau point d'équilibre qui dépendra de l'excitation appliquée. Pour une excitation $u_r \in [-2.5, 2.5]$, le sous système vertical se comporte comme un système **BIBO**. (**B**ounded **I**nput **B**ounded **O**utput).

Le sous système horizontal a le comportement d'un intégrateur, il a une infinité de points d'équilibres (tout le plan horizontal), et il diverge même pour de petites excitations.

II.7 VALIDATION DU MODELE DE SIMULATION

Pour valider notre modèle de simulation, nous allons comparer les réponses obtenues précédemment avec les réponses du système réel.

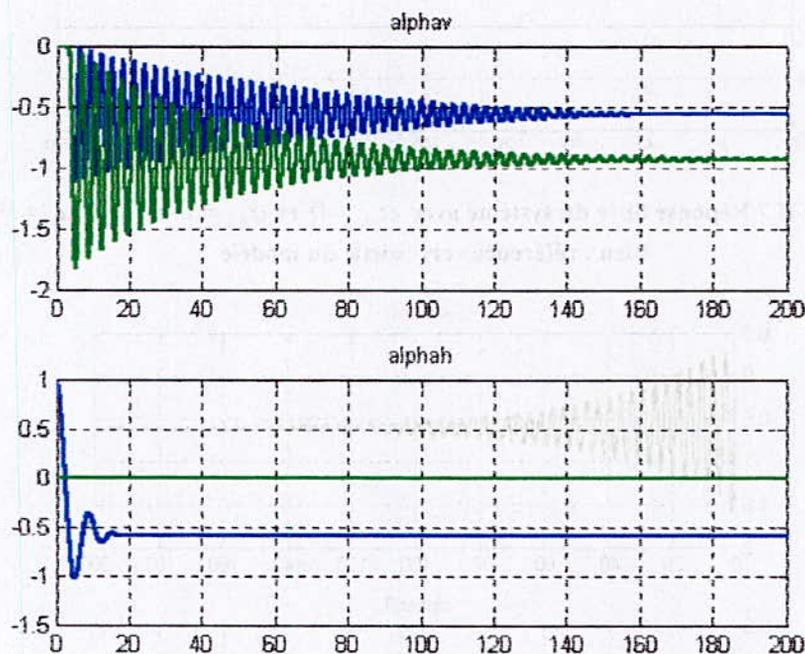


Figure II.9 : Réponse libre du système réel VS modèle de simulation
En bleu : système réel et en vert : modèle de simulation

On constate une différence entre les deux réponses, qui sera corrigé en précédant par une identification statique des paramètres.

II.7.1 Identification statique [bald02]

- Considérons d'abord le modèle vertical :

A l'équilibre nous avons :

$$\begin{cases} \dot{x}_1 = x_2 = 0 \\ \dot{x}_2 = x_3 = 0 \\ \dot{x}_3 = x_4 = 0 \end{cases} \quad (\text{II.38})$$

Ce qui donne :

$$g((A-B)\cos(x_1) - C\sin(x_1)) = 0 \quad (\text{II.39})$$

A fin d'atteindre le même régime permanent nous avons ajouté une force de torsion pour déplacer le système au nouveau point d'équilibre.

L'équation devient alors :

$$g((A-B)\cos(\alpha_{v0}) - c\sin(\alpha_{v0})) + F = 0 \quad (\text{II.40})$$

Avec

$$F = f_v \alpha_{v0} \quad (\text{II.41})$$

Où :

α_{v0} : Est la position d'équilibre du système réel (dans ce cas $\alpha_{v0} = -0.5645 \text{ rad}$)

f_v : Est une constante, déterminée à partir des équations (II.40) et (II.41)

$$f_v = \frac{g((A-B)\cos(\alpha_{v0}) - c\sin(\alpha_{v0}))}{\alpha_{v0}} \quad (\text{II.42})$$

$$\text{A.N } f_v = 0.1227 \text{ [N.m}^2.\text{rad}^{-1}\text{]}$$

Détermination de k_v coefficient de frottement

Lorsque les moteurs sont désactivés on a :

$$\begin{cases} \dot{x}_3 = x_4 = 0 \\ \dot{x}_4 = \frac{1}{J_v} x_2 \\ \dot{x}_2 = \frac{k_v}{J_v} x_2 + g((A-B)\cos(x_1) - c\sin(x_1)) + F \end{cases} \quad (\text{II.43})$$

Après intégration du nouveau système on obtient \hat{x}_1 la valeur prédite de l'angle x_1

On forme alors un problème d'identification des moindres carrés

$$J = \sum_{i=1}^{N-1} (x_1(i\Delta t) - \hat{x}_1(i\Delta t))^2 \quad (\text{II.44})$$

Avec

x_1 est le vecteur de la réponse libre du système réel et \hat{x}_1 et le vecteur de la réponse libre prédite du modèle de simulation .

Puisque tous les autres paramètres sont connus à savoir J_v, A, B et C ainsi que F , il reste alors à déterminer k_v qui minimise le critère J . Ainsi la nouvelle valeur est :

$$k_v = 0.0023$$

- **Modèle horizontal**

De même que pour le modèle vertical, on va adapter le point d'équilibre ainsi que le coefficient de friction.

Nous remarquons d'après la figure [II.9] que le système se stabilise a -0.54 rad quelque soit sa position initiale. A l'inverse du modèle de simulation. Cela est dû à une force de torsion négligée lors de la modélisation, dont l'origine est les nappes des capteurs.

La force de torsion est de la forme $M'_{th} = f_h(\alpha_h - \alpha_{h0})$.

α_{h0} : Est la position d'équilibre du système réel. (dans ce cas $\alpha_{h0} = -0.54 \text{ rad}$)

f_h : Est le coefficient de torsion.

Adaptation de k_h

De la même façon que pour le vertical on obtient k_h en minimisant le critère J ,

la nouvelle valeur de k_h est

$$k_h = 0.015$$

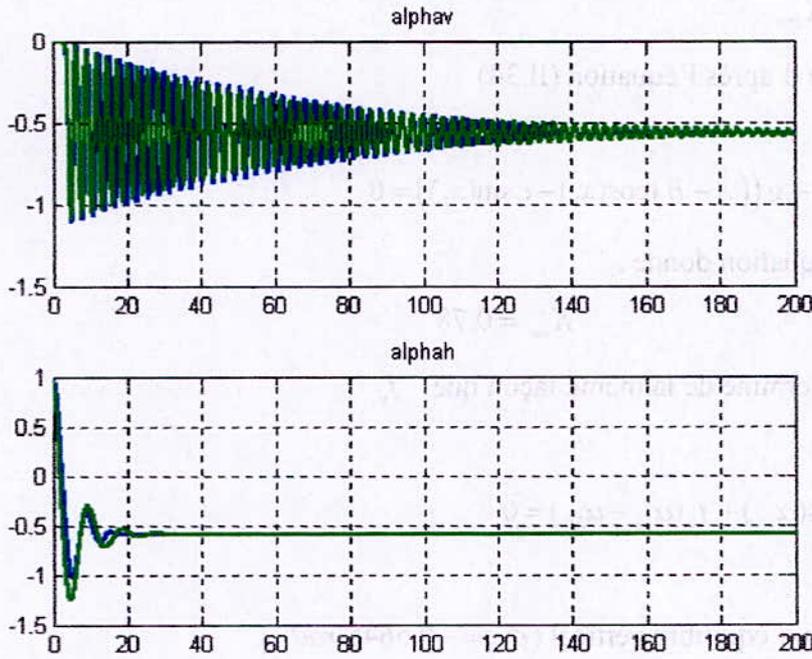


Figure II.10 : réponse après adaptation de k_h

Détermination de K_{mr} et f_h

On applique en premier lieu une tension de commande $u_v = 1.2V$ et $u_h = 0.5V$. On la comparant avec celle du modèle réel, nous constatons une erreur entre les deux modèles significative.

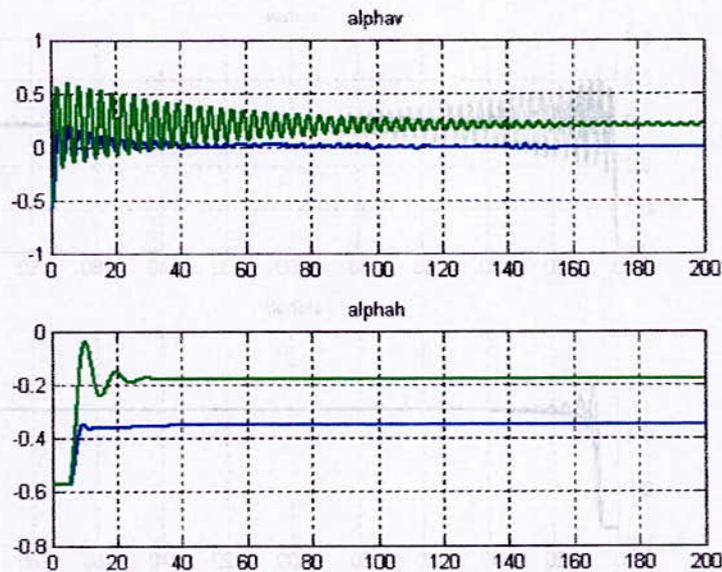


Figure II.11 : réponse avec excitation des moteurs avant l'adaptation
En bleu système réel et en vert : modèle de simulation

Nous avons alors adapté le gain statique du moteur principal K_{mr} pour le vertical et f_h pour l'horizontal

Détermination K_{mr}

On a, à l'équilibre d'après l'équation (II.34)

$$\begin{cases} x_3 = K_{mr} u_v \\ l_m F_v(P_v(K_{mr} u_v)) + g((A-B)\cos(x_1) - c\sin(x_1)) = 0 \end{cases} \quad (\text{II.45})$$

La solution de l'équation donne :

$$K_{mr} = 0.78$$

Pour f_h , il est déterminé de la même façon que f_v .

A l'équilibre on a :

$$l_h F_h(P_h(k_r u_h)) \cos(\alpha_{v0}) + f_h(\alpha_{hl} - \alpha_{h0}) = 0 \quad (\text{II.46})$$

Où :

α_{v0} : Le position d'équilibre vertical ($\alpha_{v0} = -0.5645 \text{ rad}$).

α_{h0} : La position d'équilibre du système réel. (dans ce cas $\alpha_{h0} = -0.54 \text{ rad}$).

α_{hl} : Le nouveaux position d'équilibre ($\alpha_{hl} = -0.35 \text{ rad}$).

Et cela donne :

$$f_h = -\frac{l_h F_h(P_h(k_r u_h)) \cos(\alpha_{v0})}{(\alpha_{hl} - \alpha_{h0})} \quad (\text{II.47})$$

A.N : $f_h = 0.092$

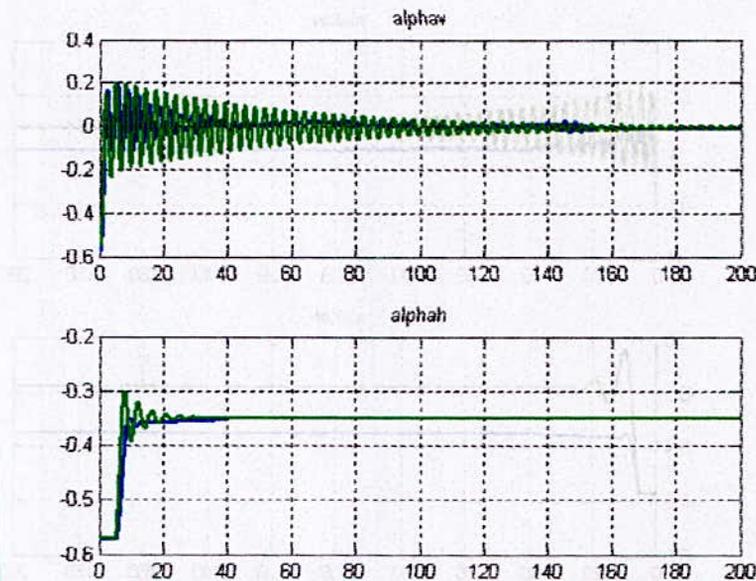


Figure II.12 : Réponse après adaptation

II.8 CONCLUSION

Nous avons procédé par la modélisation analytique pour avoir un modèle mathématique du TRMS, un modèle qui est non linéaire et couplé. Bien que nous avons essayé d'adapter le modèle au système réel, une erreur de modélisation due essentiellement aérodynamique négligées persiste. (exemple : problème des nappes des capteurs)

CHAPITRE III

RAPPELS THEORIQUES

CHAPITRE

3

III.2. COMMANDE PAR LOGIQUE FLUQUE

III RAPPELS THEORIQUES**III.1 INTRODUCTION**

La théorie de la commande est une branche de la théorie des systèmes dynamiques. Son but premier est de manipuler les variables d'entrée $u(t)$ d'un système donné et manière à *asservir* les variables de sortie $y(t)$ à une trajectoire de référence $y_r(t)$, appelée consigne [SepOI].

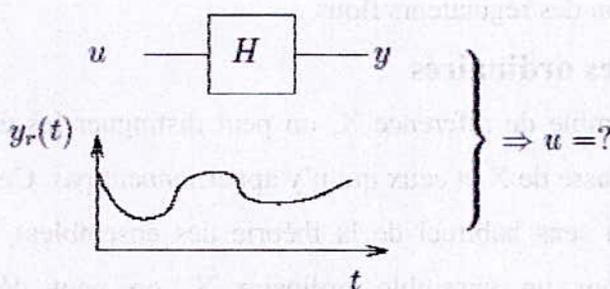


Figure III.1 : Déduire l'entrée en fonction d'une sortie de référence

D'un point de vue mathématique, il s'agit d'un problème d'inversion. Si un modèle du système $y = Hu$ est connu avec grande précision, le calcul d'un modèle inverse H^{-1} permet le calcul de l'entrée $u_r = H^{-1}y_r$ à appliquer au système pour obtenir un asservissement parfait [SepOI].

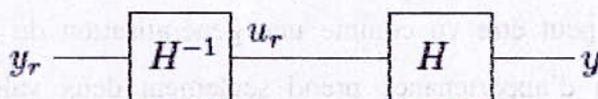


Figure III.2 : Inversion du système pour déduire l'entrée

Dans ce chapitre nous allons introduire des rappels théoriques sur les différentes techniques de commandes non linéaires qu'on va appliquer par la suite à notre système. Nous commencerons par la commande par logique floue, en suite la commande par découplage non linéaires et enfin la commande à structure variable.

III.2 COMMANDE PAR LOGIQUE FLOUE

C'est à Lotfi Zadeh, spécialiste renommé en automatique et en théorie des systèmes, à l'université de Berkeley, que revient le mérite d'avoir établi les bases théoriques de la logique floue. L'intérêt de la logique floue réside dans sa capacité à traiter l'imprécis, l'incertain et le vague. Elle est issue de la capacité de l'homme à décider et agir de façon pertinente malgré le flou des connaissances disponibles. En effet, la logique floue a été introduite pour approcher le raisonnement humain à l'aide d'une représentation adéquate des connaissances.

III.2.1 Notion de base sur la logique floue [KOS 92][buh94]

Dans cette section, nous allons voir les concepts de la théorie de la logique floue nécessaires pour la conception des régulateurs flous.

III.2.1.1 Ensembles ordinaires

Etant donné un ensemble de référence X , on peut distinguer les éléments de X qui appartiennent une certaine classe de X et ceux qui n'y appartiennent pas. Cette classe est alors un sous ensemble de X (au sens habituel de la théorie des ensembles), on le qualifie de classique ou d'ordinaire. Pour un ensemble ordinaire X , on peut définir la fonction d'appartenance $\mu_X(x)$ telle que :

$$\mu_X(x) = 0 \quad \text{si } x \in X \quad (\text{III.1})$$

III.2.1.2 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 μ_A prenant ses valeurs dans l'intervalle $[0,1]$, alors l'ensemble flou A dans X est défini par l'ensemble des paires (élément générique, degrés d'appartenance).

$$A = \{(x, \mu_x(x)) / x \in X\} \quad (\text{III.2})$$

III.2.1.3 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'appartenance.

III.2.1.4 Fonction d'appartenance [Jim93]

Afin de permettre un traitement numérique des variables linguistiques dans la prise de décisions douces sur calculateur, 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.

$$\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \quad (\text{III.3})$$

- **Fonction trapézoïdale**

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

$$\mu(x) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \quad (\text{III.4})$$

- **Fonction gaussienne**

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

$$\mu(x) = \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right) \quad (\text{III.5})$$

- **Fonction sigmoïde**

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

$$\mu(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (\text{III.6})$$

III.2.2 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 :

$$\text{" SI } x_1 \text{ est } A_1 \text{ ET } x_2 \text{ est } A_2 \text{ ALORS } x_3 \text{ est } A_3 \text{"}$$

où : " x_1 est A_1 ET x_2 est A_2 " est la prémisse de la règle

et "x3 est A3" 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*).

III.2.3 Commande par la logique floue [buh94]

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.

III.2.3.1 Contrôleur flou

La configuration de base d'un contrôleur flou (fig. III.2) 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 maximas :*

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

- *Méthode du centre de gravité:*

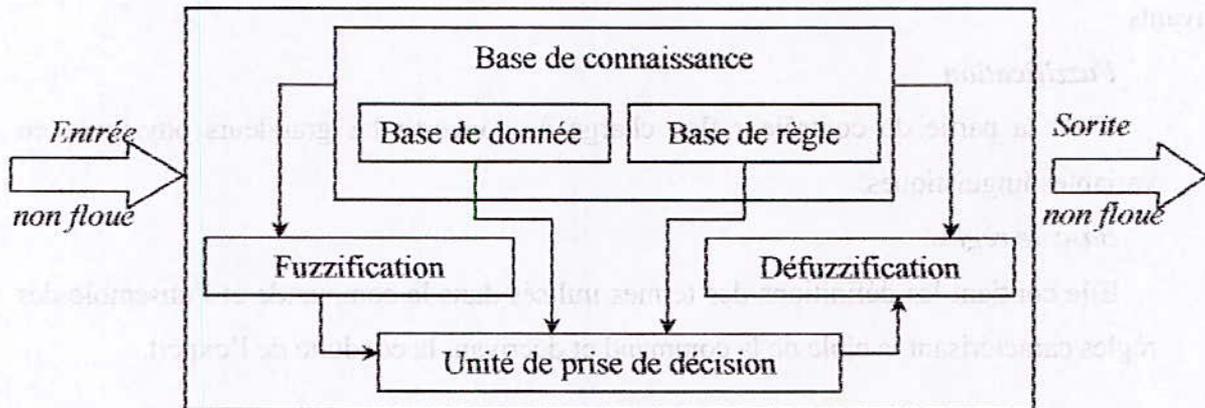
C'est la méthode la plus utilisée dans les contrôleurs flous. Dans celle-ci la commande sera égale au centre de gravité de l'ensemble flou de sortie, on obtient donc pour :

- **Univers de discours discret**

$$C = \frac{\sum_{i=1}^n x_i \mu_A(x_i)}{\sum_{i=1}^n \mu_A(x_i)} \quad (\text{III.7})$$

- **univers de discours continu**

$$C = \frac{\int x \mu_A dx}{\int \mu_A dx} \quad (\text{III.8})$$



Figur.III-3 : Architecture générale d'un contrôleur flou

III.2.4 Types de régulateurs flous

Il existe plusieurs types de régulateurs flous, qu'il diffèrent de mécanisme d'inférence utilisé, dont on cite: régulateur de Mamdani, de Sugeno, ... etc.

III.2.4.1 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 graphc flou associé à chaque règle et l'opérateur maximum pour l'agrégation. Dans la règle i :

Si x_1 est A_1 et ... 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.

III.2.4.2 Régulateur de type Sugeno

Dans les régulateurs de ce type, les conclusions des règles ne sont 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 :

$$y = \frac{\sum_{i=1}^n \alpha_i(x) * b^i}{\sum_{j=1}^n \alpha_j(x)} \quad (\text{III.9})$$

où les α_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.

III.2.5 Procédé lors de la conception d'un réglage par logique floue

Le procédé à suivre lors de la conception d'un réglage par logique floue est assez différent de celui d'un réglage conventionnel. La figure III.4 en montre les étapes principales.

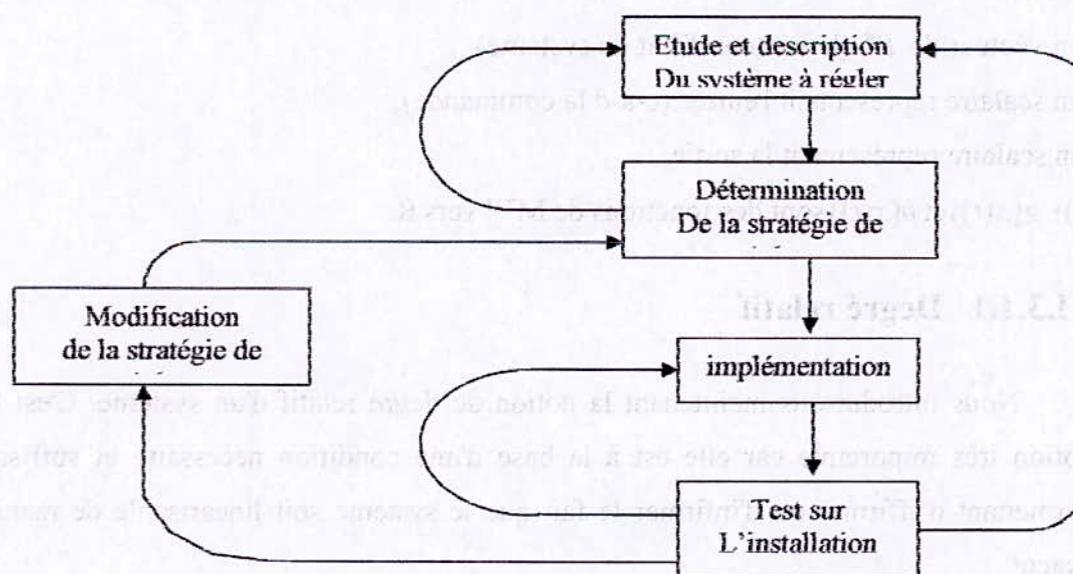


Figure III.4 : Procédé lors de la conception d'un réglage par logique floue.

III.3 COMMANDE PAR BOUCLAGE NON LINEAIRE

La commande par bouclage non linéaire est basée sur une régulation d'état permettant de transformer ce dernier en un système linéaire, constitué d'une chaîne d'intégrateurs pour chaque sortie. Il s'agit d'une linéarisation exacte du système, par opposition à l'approximation linéaire qui consiste à négliger les termes d'ordre supérieur à un, pour un vecteur d'état confiné dans un voisinage proche d'un état d'équilibre.

Nous allons présenter que les notions de linéarisation de système non-linéaire dans le cadre des systèmes ne comportant qu'une seule entrée et une seule sortie (dénommés "systèmes SISO").

III.3.1 Théorie de la linéarisation des systèmes SISO non linéaires [isid03]

Soit le système non linéaire suivant :

$$\begin{cases} \dot{x} = f(x(t)) + g(x(t))u(t) \\ y = h(x(t)) \end{cases} \quad (III.10)$$

- x est un vecteur de \mathbb{R}^n (le vecteur d'état du système),
- u est un scalaire représentant l'entrée (c-à-d la commande),
- y est un scalaire représentant la sortie,
- $f(x(t))$, $g(x(t))$ et $h(x(t))$ sont des fonctions de \mathbb{M}^{TM} vers \mathbb{R} .

III.3.1.1 Degré relatif

Nous introduisons maintenant la notion de degré relatif d'un système. C'est une notion très importante car elle est à la base d'une condition nécessaire et suffisante permettant d'affirmer ou d'infirmer le fait que le système soit linéarisable de manière exacte.

Définition

Le degré relatif (noté r) d'un système SISO peut être défini de manière intuitive comme étant le nombre minimum de fois qu'il faut dériver par rapport au temps l'expression de la sortie (y) pour voir apparaître explicitement l'entrée (u) :

$$\left\{ \begin{array}{l} y(x) = h(x) \\ y^{(1)}(x) = h_1(x) \\ y^{(2)}(x) = h_2(x) \\ \vdots \\ y^{(r-1)}(x) = h_{r-1}(x) \\ y^{(r)}(x) = a(x) + b(x)u \end{array} \right. \quad (III.11)$$

Remarque

Le degré relatif n'est pas défini en les points x de l'espace d'état tels que $a(x) = 0$.

III.3.1.2 Forme normale

Nous introduisons ensuite la notion de forme normale d'un système non-linéaire. Celle-ci est basée sur la définition d'un nouveau vecteur d'état au moyen d'un changement de variables. Ce nouveau vecteur d'état permet d'exprimer les équations d'état du système sous une forme considérablement plus simple que celle de départ. Lorsque par la suite, nous devons appliquer une régulation d'état au système, il sera plus aisé d'utiliser les équations exprimées en forme normale pour illustrer les conséquences engendrées par celle-ci.

Changement de variable

Pour tout point x de l'espace d'état en lequel le degré relatif r est défini, nous pouvons définir un nouveau vecteur d'état z que nous relierons à x par un changement de variable : $z = \varphi(x)$. Les r premières composantes de $\varphi(x)$ sont définies comme étant les dérivées d'ordre 0 à $(r-1)$ de la sortie :

$$\left\{ \begin{array}{l} \varphi_1(x) = y(x) \\ \varphi_2(x) = y^{(1)}(x) \\ \varphi_3(x) = y^{(2)}(x) \\ \vdots \\ \varphi_r(x) = y^{(r-1)}(x) \end{array} \right. \quad (III.12)$$

Il est démontré dans [Isid03] que $r \leq n$ et que ces r fonctions qualifient un changement de variable partiel régulier. Il est également démontré que si $r < n$, il est possible de trouver des fonctions $\varphi_r, \varphi_{r-1}, \dots, \varphi_1$ telles que le changement de variable complet soit régulier. De plus, il est possible d'assurer que l'expression de leur dérivée temporelle ne fasse pas apparaître l'entrée explicitement.

Equations d'état

Pour chacune des $(r-1)$ premières composantes du vecteur d'état, sa dérivée est égale à la composante suivante du vecteur d'état. Autrement dit :

$$\begin{cases} \dot{z}_1 = y^{(1)}(x) = z_2 \\ \dot{z}_2 = y^{(2)}(x) = z_3 \\ \vdots \\ \dot{z}_{r-1} = y^{(r-1)}(x) = z_r \end{cases} \quad (\text{III.13})$$

Pour la composante d'ordre r , étant donné (III.13), nous obtenons :

$$\dot{z}_r = y^{(r)}(x) = a(x) + b(x)u \quad (\text{III.14})$$

Vu que le changement de variable est régulier, nous en déduisons :

$$\begin{aligned} \dot{z}_r &= a(\varphi^{-1}(z)) + b(\varphi^{-1}(z))u \\ &= a_z(z) + b_z(z)u \end{aligned} \quad (\text{III.15})$$

Avec $a_z(z) \neq 0$ par définition du degré relatif.

Si les $\varphi_{r+1}, \dots, \varphi_n$ ont été choisies telles que leur dérivée ne fasse pas apparaître u , nous obtenons pour $r+1 < i < n$:

$$\dot{z}_i = q_i(z) \quad (\text{III.16})$$

Forme normale

Les équations d'états prennent donc la forme :

$$\left\{ \begin{array}{l} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \vdots \\ \dot{z}_{r-1} = z_r \\ \dot{z}_r = a_z(z) + b_z(z)u \\ \dot{z}_{r+1} = q_1(z) \\ \vdots \\ \dot{z}_n = q_n(z) \end{array} \right. \quad (\text{III.17})$$

La sortie s'exprime simplement par $y = z_1$.

La structure de ces équations est illustrée dans la figure III.5. Le système représenté sous cette forme est dit représenté en *forme normale*.

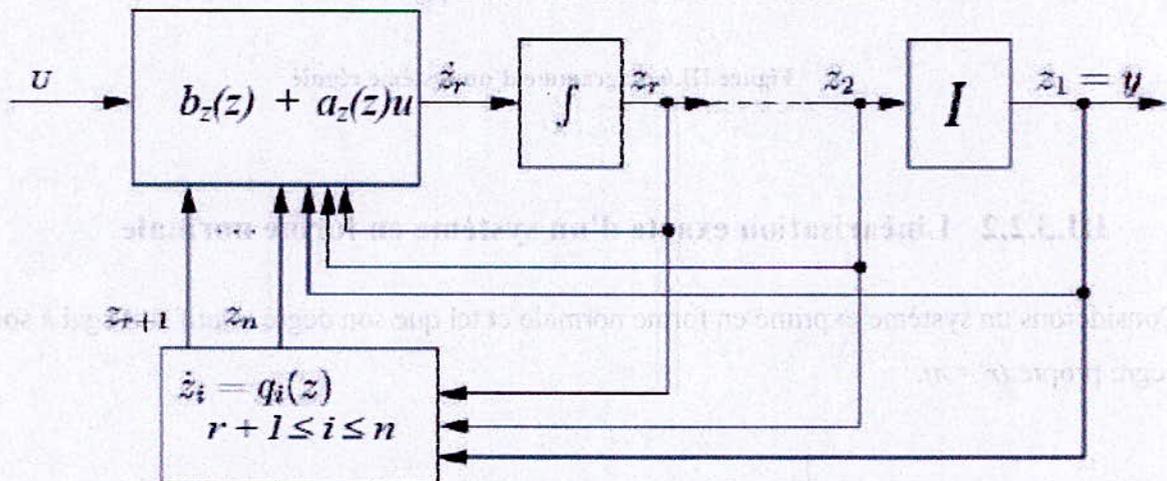


Figure III.5 : Schéma d'un système SISO en forme normale

III.3.2 Linéarisation exacte par régulation d'état statique [slot92]

III.3.2.1 Définition :

La régulation d'état statique consiste à appliquer au système une entrée ne dépendant que de son vecteur d'état actuel (x) et d'une entrée de référence (v) :

$$u = \alpha(x) + \beta(x)v$$

Le système régulé est donc de la forme :

$$\begin{cases} \dot{x} = f(x) + g(x)\alpha(x) + g(x)\beta(x)u(t) \\ y = h(x(t)) \end{cases} \quad (\text{III.18})$$

La figure III.6 illustre le bloc diagramme d'un système régulé.

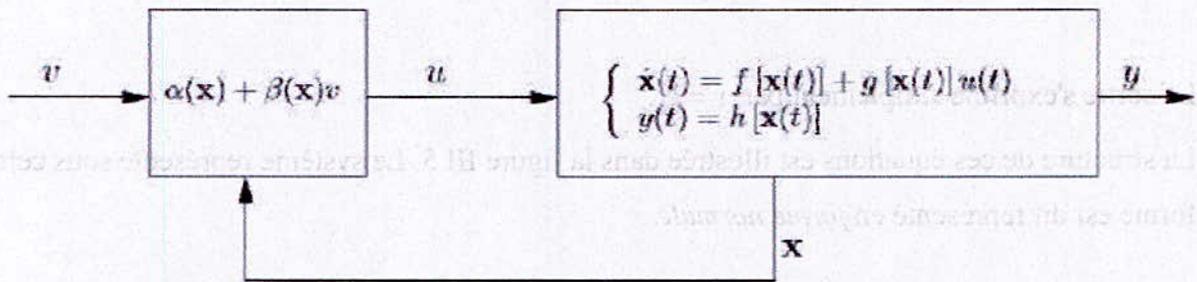


Figure III.6 diagramme d'un système régulé

III.3.2.2 Linéarisation exacte d'un système en forme normale

Considérons un système exprimé en forme normale et tel que son degré relatif soit égal à son degré propre ($r = n$).

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \vdots \\ \dot{z}_{n-1} = z_n \\ z_n = a_z(z) + b_z(z)u \end{cases} \quad (\text{III.19})$$

Nous avons vu dans l'équation (III.15) que $a_z(z) \neq 0$. Nous pouvons donc appliquer l'entrée suivante au système :

$$u(z) = \frac{1}{a_z(z)} [-b_z(z) + v] \quad (\text{III.20})$$

Les équations d'états du système régulé sont donc :

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \vdots \\ \dot{z}_{n-1} = z_n \\ \dot{z}_n = v \end{cases} \quad (\text{III.21})$$

La figure III.7 illustre le bloc diagramme correspondant.

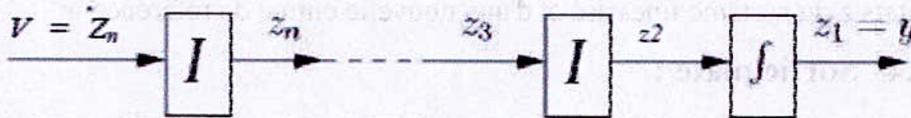


Figure III.7: Schéma du système linéarisé

Le système régulé peut donc être considéré comme une chaîne d'intégrateurs, c'est-à-dire un système linéaire et commandable. En assignant la dérivée d'ordre n de la sortie de référence (y_r) à l'entrée de référence (v), nous obtenons bien la sortie de référence comme sortie du système

$$(y = y_r).$$

Remarque

Nous venons de voir que la condition $r = n$ est **suffisante** à la linéarisation exacte du système. Il est également démontré dans [IsidO3] que cette condition est **nécessaire**.

III.3.2.3 Linéarisation exacte d'un système en forme quelconque

La linéarisation de la section précédente repose sur deux opérations :

- un changement de variable $z = \varphi(z)$.

- une régulation d'état.

Il est démontrable (voir [Isid03]) que l'ordre de ces deux opérations n'a pas d'importance. De plus, il n'est pas nécessaire de passer par la forme normale pour calculer l'expression de u . En effet, nous pouvons calculer directement $u(x)$ à partir de l'expression de $y^{(r)}$ (voir équation (2.10) dans le cas $r = n$:

$$u(x) = \frac{1}{b(x)} [-a(x) + v] \quad (\text{III.22})$$

Remarque sur le système linéarisé

Le système linéaire obtenu suite à la régulation d'état consiste en une chaîne d'intégrateurs. Remarquons que cela ne constitue pas un système **BIBO** (**B**ounded **I**ntput, **B**ounded **O**utput) stable. En effet, si nous appliquons une consigne constante en entrée, la sortie sera proportionnelle au temps élevé à une puissance égale au nombre d'intégrateurs de la chaîne.

Le système obtenu est donc instable et par conséquent difficilement utilisable. Il peut être stabilisé au moyen d'une régulation d'état supplémentaire consistant à exprimer l'entrée v en fonction des états z du système linéarisé et d'une nouvelle entrée de référence w .

III.3.2.4 Sortie plate :

Nous avons vu dans les sections précédentes que la condition principale de linéarisation d'un système était liée à son degré relatif. Ce dernier dépend quant à lui de la sortie du système, vu sa définition. La satisfaction de la condition de linéarisation est donc liée au choix de la sortie.

Jusqu'à présent, nous avons considéré que la fonction $h(x)$, autrement dit la sortie du système, était une donnée du problème. En pratique, il n'en est rien. En effet, les lois de la physique nous permettent de décrire l'évolution de l'état du système en fonction de son état actuel et des entrées qui lui sont appliquées. Elles ne nous donnent en revanche aucune indication quant au choix de la fonction $h(x)$.

Nous avons donc la liberté de choisir une sortie qui nous arrange, c'est-à-dire dans notre cas, une sortie telle que la condition $r = n$ soit respectée. Une telle sortie est appelée *sortie plate* du système. Il n'existe pas de mécanisme déterministe permettant de trouver une sortie plate pour un système donné. La recherche d'une sortie plate est donc basée sur l'intuition et la bonne connaissance du système étudié.

III.4 COMMANDE A STRUCTURE VARIABLE

Les lois de commande classique tel que PID donnent de bons résultats dans le cas des systèmes linéaires à paramètres constants. Pour des systèmes non linéaires ou ayant des paramètres non constants, ces lois de commande classique peuvent être insuffisantes car elles sont non robustes. Les lois de commande dite à structure variable constituent une bonne solution à ces problèmes. La commande à structure variable (CSV) est par nature une commande non linéaire. La caractéristique principale des systèmes à structure variable est que leur loi de commande se modifie d'une manière discontinue.

Les commutations de la commande s'effectuent en fonction des variables d'état, utilisées pour créer une "variété" ou "surface" dite de glissement dont le but est de forcer la dynamique du système de correspondre avec celle définie par l'équation de la surface. Quand l'état est maintenu sur cette surface, le système est dit en régime glissant. Ainsi tant que les conditions de glissement sont assurées, la dynamique du système reste insensible aux variations des paramètres du processus, aux erreurs de modélisation (dans une gamme qui reste plus large par rapport à celle des approches classiques de l'automatique), et à certaines perturbations.

Ce mode de glissement est souvent qualifié d'idéal du fait qu'il requiert pour exister, une fréquence de commutation infiniment grande. De fait, tout système de commande comprend des imperfections telles que retards, hystérésis, qui imposent une fréquence de commutation finie. La trajectoire d'état oscille alors dans un voisinage de la surface de glissement, phénomène appelé **chattering** ou **broutement**.

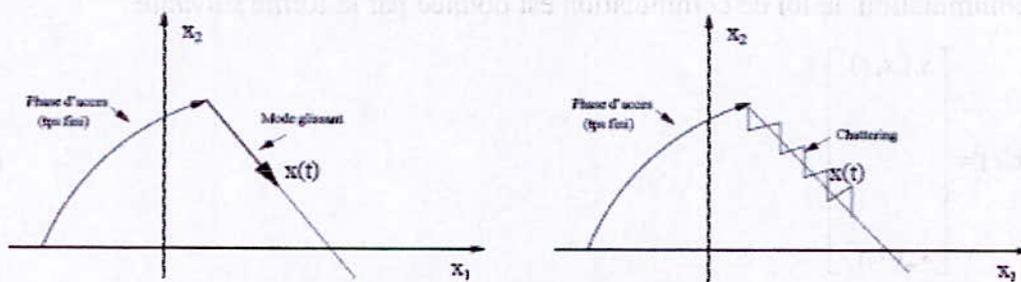


Figure III.8 : phénomène du chattering

Ce type de systèmes a en général deux modes de fonctionnement :

- Le mode non glissant, (reaching mode) ou encore mode d'accès.
- Le mode glissant, (sliding mode).

III.4.1 Le mode glissant

L'étude du régime glissant d'un système de commande à structure variable sous-entend la définition et l'étude de problèmes particuliers tels que :

- Choix de la surface de glissement.
- Existence et unicité des solutions en régime glissant (commande équivalente).
- Invariance du régime glissant vis à vis d'incertitudes paramétriques et/ou perturbations.

III.4.1.1 Choix de la surface du glissement [slot92]

Le choix de la surface de glissement concerne le nombre nécessaire ainsi que la forme de ces dernières, ces deux facteurs sont en fonction de l'application et l'objectif visé.

Soit le système défini par l'équation d'état suivante :

$$\dot{x}(t) = A(x, t) + B(x, t)u(t) \quad (\text{III.23})$$

Il faut choisir m surfaces de glissement pour un vecteur u de dimension m . En ce qui concerne la forme de la surface, deux possibilités se présentent, soit dans le plan de phase ou dans l'espace d'état, pour ce dernier, on trouve la méthode connue par « loi de commutation par retour d'état ». Celle-ci utilise les concepts de la commande par retour d'état pour synthétiser la loi de commutation, la loi de commutation est donnée par la forme suivante :

$$S(x, t) = \begin{bmatrix} s_1(x, t) \\ \vdots \\ s_m(x, t) \end{bmatrix} \quad (\text{III.24})$$

Avec :

$x \in \mathbb{R}^n$ Vecteur d'état du système.

$T \in \mathbb{R}^{m \times n}$ Matrice des gains définissant la dynamique de la surface de glissement ,
 donc :

$$s_i(x, t) = \sum_{j=1}^n T_{ij} x_j \quad (\text{III.25})$$

Dans le cas du traitement dans l'espace de phase, la fonction de commutation est une fonction scalaire, telle que la variable à régler glisse sur cette surface pour atteindre l'origine du plan de phase. Ainsi la surface $S(x)$ représente le comportement dynamique désiré du système.

J.J Slotine [slot] propose une forme d'équation générale pour déterminer la surface de glissement qui assure la convergence d'une variable x à sa valeur de consigne.

$$S(x) = \left(\frac{d}{dt} + \lambda_x \right)^{r-1} e(x) \quad (\text{III.26})$$

x : variable à commander.

$e(x)$: l'erreur de poursuite $e(x) = x - x_d$.

λ_x : Une constante positive qui interprétera la dynamique de la surface

r : le degré relatif de système.

$S(x) = 0$ est une équation différentielle linéaire dont l'unique solution est $e(x) = 0$, pour un choix convenable du gain λ_x . La linéarisation exacte de l'erreur consiste à forcer sa dynamique d'être une dynamique d'un système linéaire autonome d'ordre r .

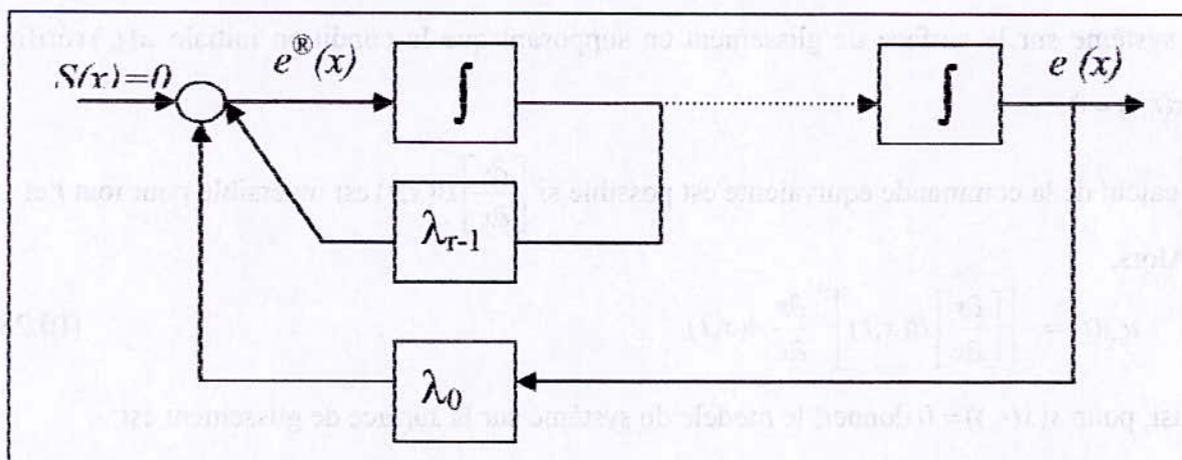


Figure III.9 : Linéarisation exacte de l'erreur

III.4.1.2 Condition d'existence du régime glissant [slot92][[buhl86][pau04]

Les systèmes de commande à structure variable sont modélisés par des équations différentielles présentant des discontinuités, (dans le second membre), du fait de la commutation de la commande. Ils ne satisfont donc pas les résultats conventionnels d'existence et d'unicité de la théorie des équations différentielles ordinaires. La question est de savoir si le système a un comportement dynamique unique quand $s(x)=0$. Différentes méthodes de prolongement par continuité ont été proposées. Toutefois, une des approches les plus anciennes et les plus formalisées mathématiquement est la méthode développée par FILIPPOV. Elle constitue une théorie mathématique systématique pour les équations différentielles avec discontinuités. Elle possède néanmoins l'inconvénient de s'appliquer au cas mono entréc. Dans le cas multi entrées, la méthode de la commande équivalente peut être considérée comme une extension formelle de cette dernière.

Pour développer cette technique, on considère le modèle d'état (III.23)

On suppose que la trajectoire d'état atteint l'hypersurface de glissement à l'instant t_0 et qu'un mode glissant existe pour $t \geq t_0$. Cela implique

$$\begin{cases} s(x) = 0 \\ \dot{s}(x) = 0 \end{cases} \quad (\text{III.27})$$

Ce qui conduit après substitution de \dot{s} à écrire :

$$\frac{\partial s}{\partial x} \dot{x} = \frac{\partial s}{\partial x} [A(x,t) + B(x,t)u_{eq}(t)] = 0$$

Où u_{eq} est la **commande équivalente** qui résout l'équation. Cette commande étant supposée connue et introduite dans l'équation du modèle, on obtient alors le modèle du comportement du système sur la surface de glissement en supposant que la condition initiale $x(t_0)$ vérifie $s(x(t_0)) = 0$

Le calcul de la commande équivalente est possible si $\left[\frac{\partial s}{\partial x} \right] B(x,t)$ est inversible pour tout t et x . Alors,

$$u_{eq}(t) = - \left[\left[\frac{\partial s}{\partial x} \right] B(x,t) \right]^{-1} \frac{\partial s}{\partial x} A(x,t) \quad (\text{III.28})$$

Ainsi, pour $s(x(t_0)) = 0$ donner, le modèle du système sur la surface de glissement est:

$$\begin{cases} \dot{x}(t) = \left[I - B(x,t) \left[\left[\frac{\partial s}{\partial x} \right] B(x,t) \right]^{-1} \frac{\partial s}{\partial x} \right] A(x,t) \\ s(x) = 0 \end{cases} \quad (\text{III.29})$$

Il est remarquable de constater que les dynamiques du système en mode glissant sont d'ordre inférieur au système original. Cette réduction d'ordre est aisément explicable par le nombre de variables d'état contraintes par la relation $s(x) = 0$

III.4.1.3 Principe d'invariance [Wall89]

Le développement d'une forme générale de la commande à structure variable, nécessite le modèle du système, la forme générale du modèle est donné par :

$$\dot{x} = b(x) [f(x) + u + d(t)] \quad (\text{III.30})$$

L'équation (III.30) représente le modèle exact du système, qui en réalité ni jamais connu complètement. En conséquence, un modèle nominal ou approximatif du système déterminé analytiquement et/ou expérimentalement est utilisé pour développer une loi de commande. La différence entre le modèle exact et nominal a un impact direct sur les performances de la commande.

La capacité de la commande de maintenir la stabilité en présence de grandes erreurs de modélisations est appelé « stabilité robuste », pour les commandes linéaires, l'une des méthodes pour assurer la stabilité robuste est d'estimer l'amplitude et la forme de l'erreur de modélisation, et après examiner les réponses fréquentielles de la boucle fermée (modèle nominal inclus) et l'erreur de modélisation pour garantir que le critère de la stabilité robuste est toujours vérifié.

Pour le mode de glissement, la stabilité robuste vis-à-vis des erreurs de modélisation est automatiquement garantie en introduisant explicitement l'estimation des amplitudes erreurs de modélisations dans la loi de commande. Les deux, commande linéaires et commande par mode glissants, requiert l'estimation de l'erreur de modélisation, c'est comment les considérer qui diffère. Les performances se dégradent quand l'erreur augmente dans les deux cas.

Pour les modes glissants, la modélisation des erreurs ainsi que les perturbations est souvent spécifiée par une majoration comme suit : la différence entre $f(x)$ nominal et réel peut être représenté par :

$$f(x) = \hat{f}(x) + \Delta f(x) \quad (\text{III.31})$$

Où $\hat{f}(x)$ est valeur nominale. L'amplitude de l'incertitude est bornée par :

$$F(x) \geq \Delta f(x) \quad (\text{III.32})$$

L'incertitude bornée du terme $b(x)$ est spécifiée par β , avec :

$$\frac{1}{\beta} \leq \frac{\hat{b}(x)}{b(x)} \leq \beta \quad (\text{III.33})$$

Où $\hat{b}(x)$ est la valeur nominale. De même la majoration sur la perturbation, $d(t)$, est caractérisée par :

$$D(t) = |d(t)| \quad (\text{III.34})$$

Majoration de la perturbation inclus aussi, toute dynamique non modélisée, non linéarités non structurée et les variations paramétrique en dehors de $F(x)$ ou β . Les majorations sur les erreurs de modélisation et les perturbations ($F(x)$, β et $D(t)$) sont utilisés pour définir le gain K pour le mode non glissant

III.4.2 Le mode non glissant [dec88][utk77]

Le mode préliminaire au mode glissant, partant d'une condition initiale quelconque pour atteindre la surface de glissement est appelé « **attractivité** » ou **mode non glissant**, (reaching mode en anglais). La définition complète de ce mode nécessite la définition d'une **condition d'attractivité** ainsi que la définition de la loi de commande non linéaire et de sa structure.

III.4.2.1 Conditions d'attractivité

Cette condition est en fait la condition sous laquelle le mode de glissement existe et sous laquelle la trajectoire d'état va effectivement atteindre la surface de glissement en un temps fini. Deux types de conditions d'accès à la surface de glissement sont présentés.

III.4.2.2 Approche directe

Cette approche est la plus ancienne, elle a été proposée par EMILYANOV et UTKIN. Elle est globale mais ne garantit pas en revanche un temps d'accès fini.

$$\begin{cases} \dot{S}(x) > 0 \text{ lorsque } S(x) < 0 \\ \dot{S}(x) < 0 \text{ lorsque } S(x) > 0 \end{cases} \quad (\text{III.35})$$

Où bien

$$\dot{s}_i(x,t) s_i(x,t) < 0 \quad i = 1, \dots, m \quad (\text{III.36})$$

Cette condition est toutefois difficile à utiliser pour faire la synthèse de la loi de commande, Particulièrement dans le cas d'un système multi entrées.

III.4.2.3 Approche de LYAPUNOV

Il s'agit de formuler une fonction scalaire définie positive $V(x) > 0$ pour les variable d'état du système, et de choisir la loi de commutation qui fera décroître cette fonction. Elle est utilisée pour estimer les performances de la commande, l'étude de la robustesse, et pour garantir la stabilité des systèmes non linéaires.

$$\dot{V}(x) < 0 \text{ avec } V(x) > 0 \quad (\text{III.37})$$

En définissant la fonction de lyapunov par

$$V(x) = \frac{1}{2} S(x)^2 \quad (\text{I.38})$$

Pour que la fonction de lyapunov décroisse, il suffit d'assurer que sa dérivée est négative.

Ceci est vérifié si :

$$S(x) \dot{S}(x) < 0 \quad (\text{III.39})$$

Cette équation explique que le carré de la distance vers la surface, mesurée par $S(x)^2$, diminue tout le temps, contraignant ainsi la trajectoire du système à se diriger vers la surface des deux cotés. Cette condition de convergence suppose un régime de glissement idéal.

Dans le cas d'un régime glissant pratique, la condition de convergence prend la forme suivante :

$$\frac{1}{2} \frac{d}{dt} S(x)^2 \leq -\eta |S(x)| \quad (\text{III.40})$$

Où η est une constante, qui définira le temps de convergence vers la surface de glissement

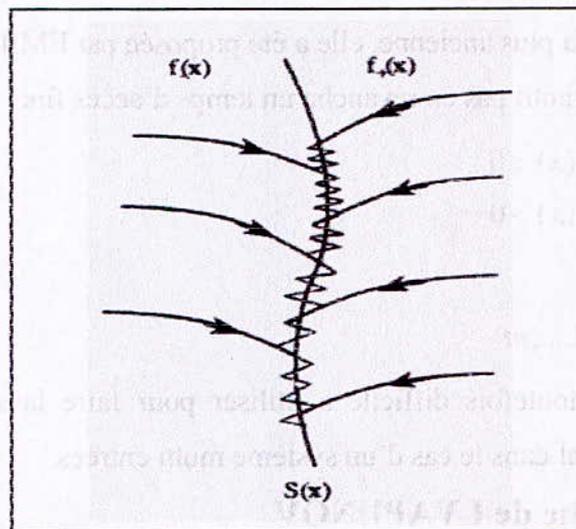


Figure III.10 Trajectoire de l'état vis-à-vis de la surface de glissement

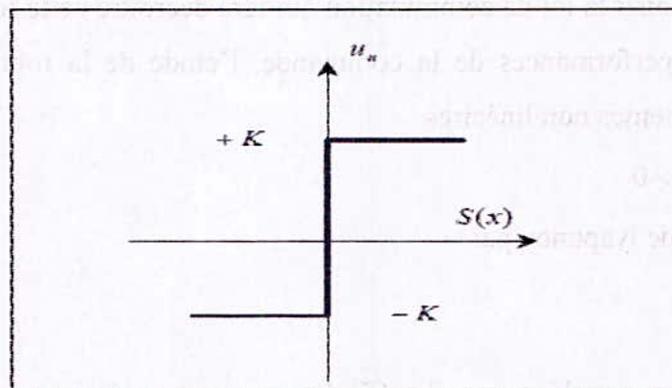
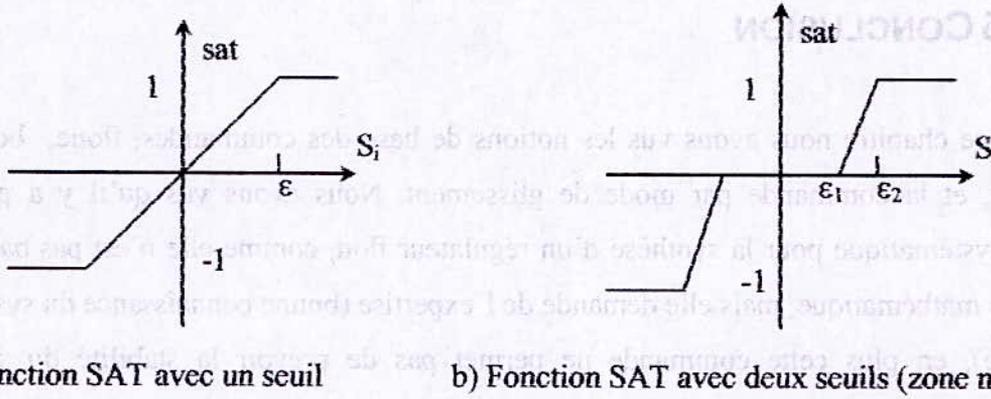


Figure III.11 Définition de la fonction signe

III.4.3 Elimination du phénomène de chattering [slot92][wall89]

Le problème des régimes glissants est que la commande est discontinue sur la surface de glissement et par conséquent, elle va commuter régulièrement ceci à chaque traversée de cette surface, ceci à une fréquence infinie, une telle commande, en plus d'être difficile à réaliser en pratique, peut ne pas être directement applicable au système. Pour remédier à ce problème, la fonction discontinue est remplacée par une fonction continue. Ceci au voisinage de la surface de glissement puis on procédera à un lissage de la commande.



a) Fonction SAT avec un seuil

b) Fonction SAT avec deux seuils (zone morte)

Figure III.12 : Fonction « SAT »

Ces deux fonctions sont définies par :

a) :

$$sat(S) = \begin{cases} -1 & \text{si } S < -\epsilon \\ \frac{S}{\epsilon} & \text{si } |S| < \epsilon \\ 1 & \text{si } S > \epsilon \end{cases}$$

b) :

$$sat(S) = \begin{cases} 0 & \text{si } |S| < \epsilon_1 \\ \frac{S - \epsilon_1}{\epsilon_2 - \epsilon_1} & \text{si } \epsilon_1 < |S| < \epsilon_2 \\ sign(S) & \text{si } |S| > \epsilon_2 \end{cases}$$

On peut aussi donner la fonction de Smooth ou sigmoïde :

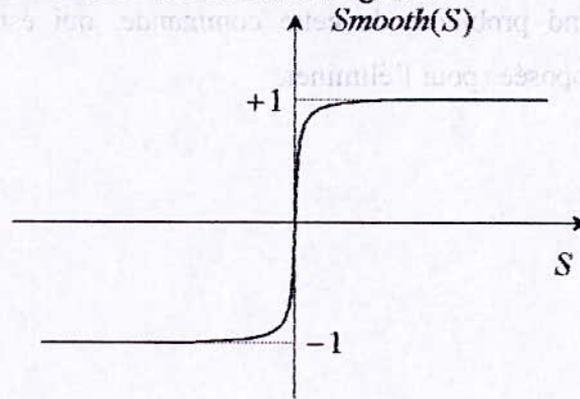


Figure 3.7 : Fonction « Smooth ».

$$Smooth(S) = \frac{S}{|S| + \mu}$$

III.5 CONCLUSION

Dans ce chapitre nous avons vu les notions de base des commandes, floue, bouclage linéarisant, et la commande par mode de glissement. Nous avons vu qu'il y a pas une méthode systématique pour la synthèse d'un régulateur flou, comme elle n'est pas basée sur un modèle mathématique, mais elle demande de l'expertise (bonne connaissance du système à commandé), en plus cette commande ne permet pas de prévoir la stabilité du système commandé.

Nous avons vu aussi la commande par bouclage linéarisant qui nous donne une linéarisation exacte du modèle. Le modèle linéaire obtenu est découplé sous forme d'une chaîne d'intégrateurs, sur lequel on peut appliquer des commandes linéaires. Autrement dit cette commande donne au système une dynamique linéaire.

En fin, nous avons vu la commande par mode de glissement, qui est une commande non linéaire robuste, elle consiste à amener le système à une surface dite de glissement, sur laquelle les objectifs de commande sont réalisables, et le maintenir sur cette surface. Nous avons vu aussi le plus grand problème de cette commande, qui est le phénomène de chattering, et les solutions proposées pour l'éliminer.



Figure 3.7 : Fonction de transfert

$$\frac{Y}{U} = \frac{1}{s+1}$$

CHAPITRE IV

SYNTHESE DES LOIS DE COMMANDES

IV SYNTHÈSE DE LA COMMANDE

IV.1 INTRODUCTION

L'objectif de ce chapitre est de synthétiser des lois de commandes afin d'asservir l'angle d'élévation et d'azimut du TRMS. En se basant sur la théorie présentée dans le chapitre III. Nous abordons d'abord la commande qualitative : par logique floue, puis nous développons une loi de commande linéarisante. En fin nous synthétiserons une commande à structure variable.

Avant l'application d'une technique de commande à un système complexe il faut choisir entre deux stratégies de commande, d'abord la commande centralisée, ou bien la commande décentralisée en s'appuyant sur les critères suivant : réalisabilité de la commande, temps de calcul et ordre du régulateur.

L'utilisation de la commande centralisée nécessite une bonne connaissance du système, donc un temps de calcul très grand. Par contre l'utilisation de la commande décentralisée facilite la synthèse et donne des lois de commandes non gourmandes en temps de calcul

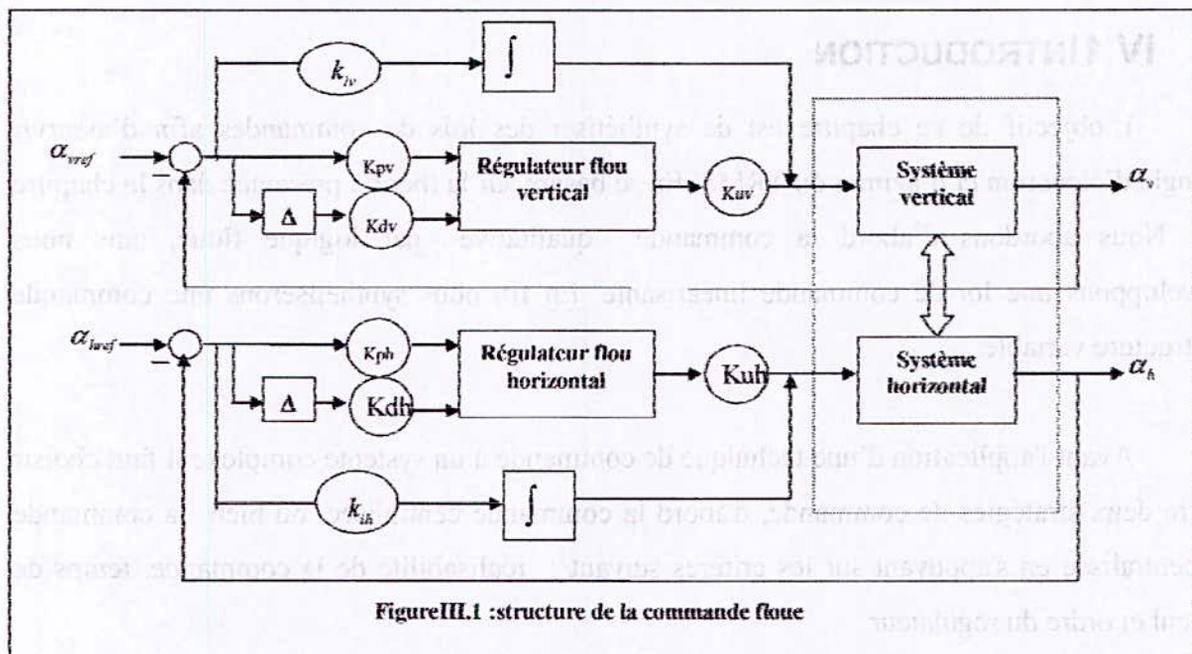
Le système global sera décomposé en deux sous-systèmes (vertical et horizontal), chacun d'eux est commandé indépendamment Les interconnexions seront considérées comme étant des perturbations.

IV.2 COMMANDE FLOUE

Cette commande a l'avantage de ne pas avoir besoin du modèle mathématique pour la synthèse, elle se caractérise par la simplicité de synthèse de la loi de commande.

IV.2.1 Structure de la commande floue [Lahd94]

Sur la figure (IV.1) sont représentés les deux régulateurs flous des deux sous-systèmes du TRMS. Ces derniers sont indépendants les un des autres, chacun à deux entrées qui sont l'erreur et la variation de l'erreur et une sortie qui est la variation de commande à appliquer au sous système correspondant.



IV.2.1.1 Régulateur flou de MAMDANI [mamd75]

Le schéma du régulateur de type MAMDANI est composé de :

- Un bloc de calcul de la variation de l'erreur (Δe) au cours du temps.
- Les facteurs d'échelle ou gains de normalisation associés à l'erreur (e), à sa variation (Δe), et à la commande (U).
- Un bloc de fuzzification de l'erreur et de sa variation.
- Une base de règles de contrôle flou.
- Une logique pour l'évaluation des règles de contrôle flou (inférence), pour notre cas c'est la méthode max-min qui est utilisée.

- Un bloc de défuzzification servant à convertir la commande floue en une valeur numérique significative.

IV.2.1.2 Loi de commande

La loi utilisée est fonction des entrées choisies pour notre contrôleur, en l'occurrence de l'erreur et sa variation sur une période d'échantillonnage T_e :

$$U(kT_e) = f(e(kT_e), \Delta e(kT_e)) \quad (IV.1)$$

Par conséquent, l'utilisation de l'ensemble des règles de décision associées donne la commande U nécessaire et qui constitue donc la sortie de notre contrôleur.

Cette commande est obtenue par lecture d'une table de décision prédéfinie hors ligne. Notre choix de la loi de commande s'est porté sur la forme suivante:

$$U(kT_e) = K_u \cdot u_n(kT_e) \quad (IV.2)$$

K_u : le gain associé à la commande.

IV.2.1.3 Les gains de normalisation

L'une des choses les plus importantes dans la commande floue est de trouver les gains de l'erreur, variation de l'erreur et de la commande ou encore gains de normalisation. Ajoutés à cela, le gain de l'actions intégrale. En effet, ce sont ces derniers qui fixeront les performances de la commande. Il n'y a aucune méthode systématique qui donne ces paramètres d'emblée ; en effet, il faut procéder par un tâtonnement en utilisant des règles empiriques et l'expérience acquise au fil du temps.

IV.2.2 Synthèse des régulateurs

La base des règles est obtenue à partir de la matrice de Macvicar-whelan, les régulateurs des deux sous-systèmes se différencient seulement par les gains de normalisation et les gains associés aux commandes, les fonctions d'appartenance sont Gaussiennes pour les entrées (erreur et variation de l'erreur) et triangulaires pour les sorties. (la commande)

$e \backslash \Delta e$	NB	NS	EZ	PS	PB
NB	NB	NB	NB	NS	EZ
NS	NB	NB	NS	EZ	PS
EZ	NB	NS	EZ	PS	PB
PS	NS	EZ	PS	PB	PB
PB	EZ	PS	PB	PB	PB

Tableau IV.1 : Base de règles, RLF (5×5)

Avec:

NB : Negative Big

NS : Negative Small

EZ : Equal Zero

PS : Positive Small

PB : Positive Big

IV.2.3 L'action intégrale

Pour éliminer l'erreur statique, on a inséré en parallèle avec le régulateur flou une action intégrale. Le régulateur global aura une structure d'un PID, dont les actions, proportionnelle et dérivée sont réalisées par un régulateur flou (PD flou).

IV.3 BOUCLAGE NON LINEAIRE

Cette commande permet de bénéficier de la théorie des systèmes linéaires avec une rigoureuse mathématique. Le fondement mathématique de cette théorie a été détaillé dans la partie rappel théorique.

Comme pour la logique floue, nous allons synthétiser deux lois de commandes linéarisante pour chaque sous-système. Ensuite nous appliquons une commande linéaire, à savoir « le placement de pole » sur les modèles obtenus

IV.3.1 Calcul du Degré relatif

Modèle vertical

$$\Sigma_v \begin{cases} \dot{x}_1 = a_v x_2 \\ \dot{x}_2 = f_h(x_3) - g_h(x_1) - b_v x_2 \\ \dot{x}_3 = -c_v x_3 + d_v u \\ y = x_1 \end{cases} \quad (\text{IV.3})$$

Avec

$$\begin{cases} f_h(x_3) = I_m F_v(P_v(x_3)) \\ g_h(x_1) = g((A-B)\cos(x_1) - c\sin(x_1)) \\ a_v = \frac{1}{J_v} \\ b_v = \frac{k_v}{J_v} \\ c_v = \frac{1}{T_{mv}} \\ d_v = \frac{k_{mv}}{T_{mv}} \end{cases} \quad (\text{IV.4})$$

Dérivons y jusqu'à l'apparition de la commande

$$\dot{y} = \dot{x}_1 = a_v x_2 \quad (\text{IV.5})$$

$$\ddot{y} = a_v \dot{x}_2 = a_v (f_h(x_3) - g_h(x_1) - b_v x_2) \quad (\text{IV.6})$$

$$\bar{y} = a_v \left(\frac{\partial f_v(x_3)}{\partial x_3} \dot{x}_3 - \frac{\partial g_v(x_1)}{\partial x_1} \dot{x}_1 - b_v \dot{x}_2 \right) \quad (\text{IV.7})$$

Remplaçons \dot{x}_1 et \dot{x}_3 par leurs valeurs

$$\bar{y} = a_v \frac{\partial f_v(x_3)}{\partial x_3} [-c_v x_3 + d_v u] - a_v^2 \frac{\partial g_v(x_1)}{\partial x_1} x_2 - a_v b_v \dot{x}_2 \quad (\text{IV.8})$$

Puisque la commande apparaît dans la 3^{ème} dérivée de y , le sous système vertical a par conséquent un degré relatif égal à 3. ce-ci nous permet de conclure que le sous-système vertical Σ_v est complètement linéarisable

$$r_v = 3 \quad (\text{IV.9})$$

Modèle horizontal

$$\Sigma_h \begin{cases} \dot{x}_1 = a_h x_2 \\ \dot{x}_2 = f_h(x_3) - b_h x_2 \\ \dot{x}_3 = -c_h x_3 + d_h u \\ y = x_1 \end{cases} \quad (\text{IV.10})$$

Tel que :

$$\begin{cases} f_h(x_3) = L_f P_h(x_3) \\ a_h = \frac{1}{J_h(\alpha_{v0})} \\ b_h = \frac{k_h}{J_h} \\ c_h = \frac{1}{T_r} \\ d_h = \frac{k_r}{T_r} \end{cases} \quad (\text{IV.11})$$

On procède de la même manière que pour le vertical,

$$\dot{y} = \dot{x}_1 = a_h x_2 \quad (\text{IV.12})$$

$$\ddot{y} = a_h \dot{x}_2 = a_h (f_h(x_3) - b_h x_2) \quad (\text{IV.13})$$

$$\ddot{y} = a_h \left(\frac{\partial f_h(x_3)}{\partial x_3} \dot{x}_3 - b_h \dot{x}_2 \right) \quad (\text{IV.14})$$

$$\ddot{y} = a_h \frac{\partial f_h(x_3)}{\partial x_3} [-c_h x_3 + d_h u] - a_h b_h \dot{x}_2 \quad (\text{IV.15})$$

De (IV.15), on conclut que Σ_h a un degré relatif égal à 3

$$r_h = 3 \quad (\text{IV.16})$$

Le sous-système horizontal Σ_h est aussi complètement linéarisable

IV.3.2 Synthèse de la loi de commande par bouclage non linéaire :

Modèle vertical

Développons (IV.8)

$$y^{(3)} = -a_v c_v \frac{\partial f_v(x_3)}{\partial x_3} x_3 + a_v d_v \frac{\partial f_v(x_3)}{\partial x_3} u - a_v^2 \frac{\partial g_v(x_1)}{\partial x_1} x_2 - a_v b_v f_v(x_3) - a_v b_v g_v(x_1) - a_v b_v^2 x_2 \quad (\text{IV.17})$$

Soit la loi de commande linéarisante suivante :

$$u_v = \frac{1}{d_v a_v \frac{\partial f_v(x_3)}{\partial x_3}} \left[-a_v c_v \frac{\partial f_v(x_3)}{\partial x_3} x_3 - a_v^2 \frac{\partial g_v(x_1)}{\partial x_1} x_2 - a_v b_v f_v(x_3) - a_v b_v g_v(x_1) - a_v b_v^2 x_2 - K_v v \right] \quad (\text{IV.18})$$

Injectons la valeur de u_v dans (IV.17)

$$y^{(3)} = K_v v \quad (\text{IV.19})$$

Ou bien

$$Y(s) = -\frac{K_v}{s^3} V \quad (\text{IV.20})$$

Cette loi de commande nous a permis d'avoir une relation linéaire entre la sortie y et la nouvelle commande v (linéarisation entrée/sortie).

Soit le changement de base suivant :

$$\begin{cases} z_1 = y_1 \\ z_2 = \dot{y}_1 \\ z_3 = \ddot{y}_1 \end{cases} \quad (\text{IV.21})$$

Après calcul, on obtient

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = z_3 \\ \dot{z}_3 = y^{(3)} = K_v v \\ y = z_1 \end{cases} \quad (\text{IV.22})$$

Ainsi le nouveau modèle obtenu est un système linéaire de la forme :

$$\begin{cases} \dot{Z} = AZ + BV \\ Y = CZ \end{cases} \quad (\text{IV.23})$$

Avec

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \text{et} \quad C = [1 \ 0 \ 0]$$

Modèle horizontal

En procédant de la même façon que pour l'horizontal, on tombe sur la commande linéarisante suivante

$$u_h = \frac{1}{d_h a_h \frac{\partial f_h(x_3)}{\partial x_3}} \left[-a_h c_h \frac{\partial f_h(x_3)}{\partial x_3} x_3 - a_h b_h f_h(x_3) - a_h b_h^2 x_2 - K_h v \right] \quad (\text{IV.24})$$

Le nouveau système linéaire est alors :

$$\begin{cases} \dot{Z} = AZ + BV \\ Y = CZ \end{cases} \quad (\text{IV.25})$$

Avec

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \text{et} \quad C = [1 \ 0 \ 0]$$

Remarques :

- La condition :

$$\frac{\partial f(x_3)}{\partial x_3} \neq 0 \quad (\text{IV.26})$$

Est nécessaire dans les deux cas (vertical et horizontal) pour l'existence d'une commande linéarisante.

- Avec le modèle linéaire obtenu, on peut synthétiser des lois de commande linéaire, (notre cas avec la commande par placement de pôles), comme on peut synthétiser des lois de commande non linéaire.

IV.3.3 Simulation

Les figures (IV.2) et (IV.3) représentent les résultats de l'application de la commande par bouclage linéaire avec placement de pôles sur les deux modèles à un degré de liberté vertical et horizontal. On constate que celle-ci a réussi à stabiliser le système, avec une tension de commande admissible par les actionneurs.

On constate aussi que les mêmes régulateurs appliqués au modèle à deux degré de liberté ont pu rejeter les effets de couplages et stabiliser le système global figure (IV.4)

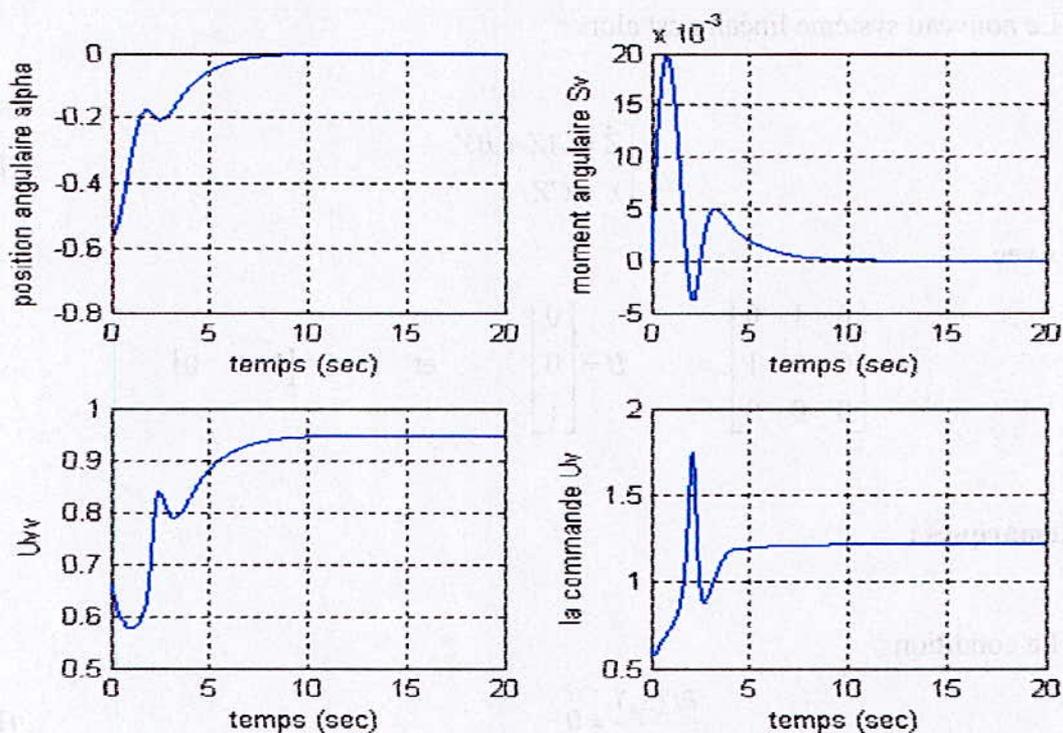


Figure IV.2 : modèle vertical 1DDL , pôles= $[-2 -2.5 -3]$

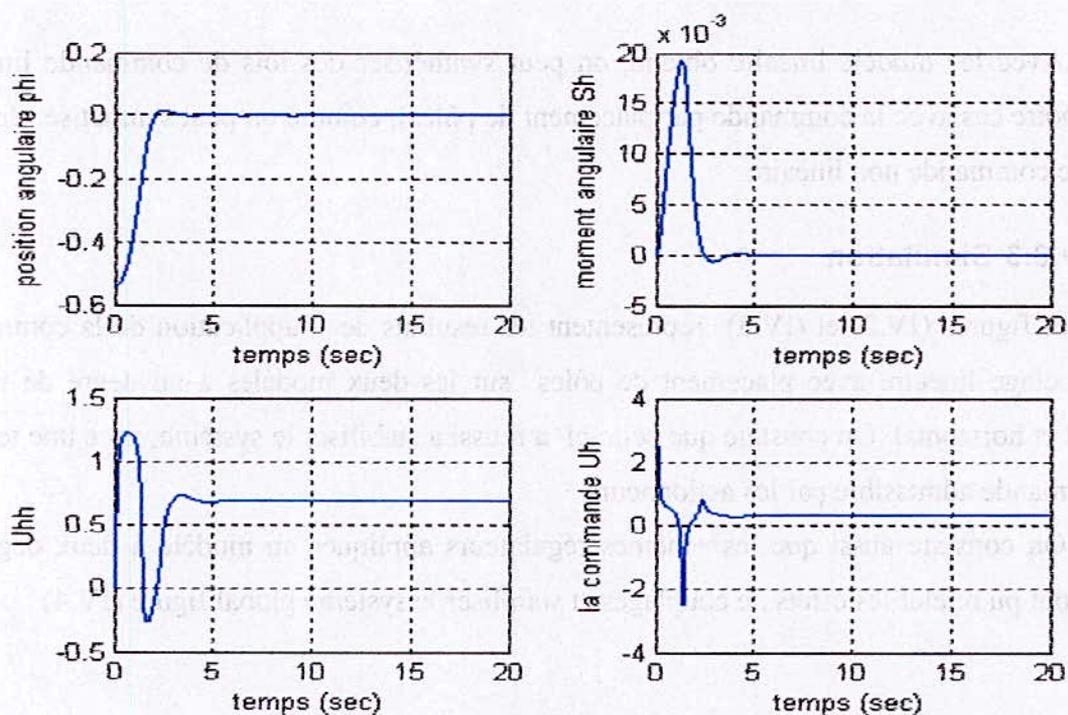


Figure IV.3 : modèle horizontal 1DDL , pôles= $[-1 -1.5 -2]$

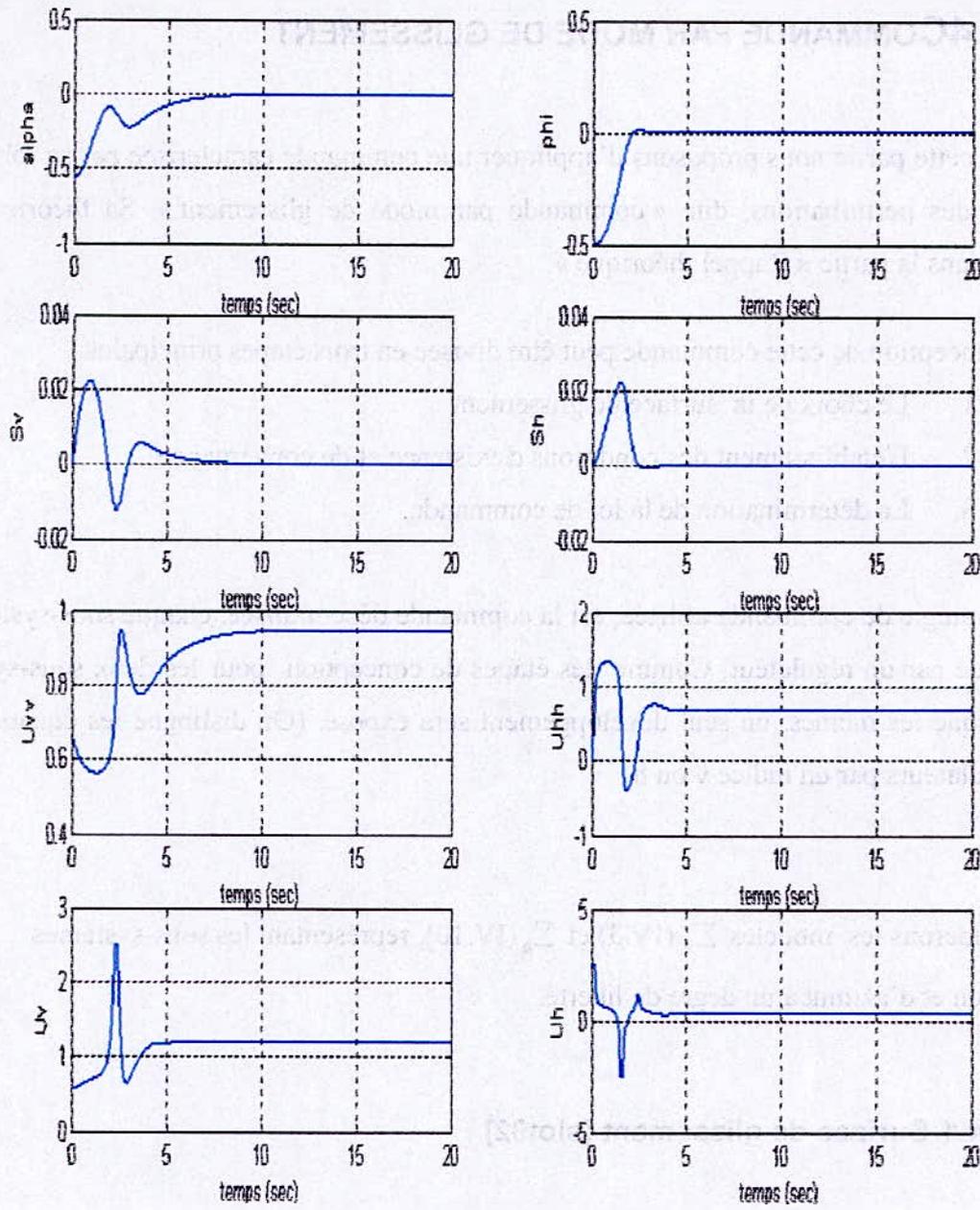


Figure IV.4 : modèle couplé 2DDL

IV.4 COMMANDE PAR MODE DE GLISSEMENT

Dans cette partie nous proposons d'appliquer une commande caractérisée par sa robustesse vis-à-vis des perturbations, dite « commande par mode de glissement ». Sa théorie a été exposée dans la partie « Rappel théorique ».

La conception de cette commande peut être divisée en trois étapes principales :

1. Le choix de la surface de glissement .
2. L'établissement des conditions d'existence et de convergence.
3. La détermination de la loi de commande.

La stratégie de commande utilisée, est la commande décentralisée, chaque sous-système est commandé par un régulateur. Comme les étapes de conception pour les deux sous-systèmes sont presque les mêmes, un seul développement sera exposé. (On distingue les équations des deux régulateurs par un indice v ou h)

Considérons les modèles Σ_v (IV.3) et Σ_h (IV.10), représentant les sous-systèmes d'élévation et d'azimut à un degré de liberté.

IV.4.1 Surface de glissement [slot92]

La surface de glissement choisie est celle proposée par J.J.E Slotine, sa forme générale est :

$$S = \left(\frac{d}{dt} + \lambda\right)^{r-1} e(x) \quad (\text{IV.27})$$

Avec

λ : Constante positive caractérisant la dynamique de la surface de glissement.

r : Degré relatif.

e : Erreur de poursuite ($e = x - x_d$) avec x_d est le trajectoire désiré).

$s = 0$ Représente une équation différentielle, dont la solution est de la forme

$$e = p(t) \exp(-\lambda t) \quad (\text{IV.28})$$

Les objectifs de poursuite sont bien réalisables sur cette surface, puisque

$$\lim_{t \rightarrow \infty} e = 0 \quad (\text{IV.29})$$

Donc

$$x \rightarrow x_d \quad (\text{IV.30})$$

Pour notre système, on a deux surfaces

$$S_v = \left(\frac{d}{dt} + \lambda_v \right) r^{-1} e_v \quad (\text{IV.31})$$

$$S_h = \left(\frac{d}{dt} + \lambda_h \right) r^{-1} e_h \quad (\text{IV.32})$$

Comme le degré relatif est 3 pour les deux sous-systèmes (IV.9) et (IV.16), on obtient

$$S_v = \lambda_v^2 e_v + 2\lambda_v \dot{e}_v + \ddot{e}_v \quad (\text{IV.33})$$

$$S_h = \lambda_h^2 e_h + 2\lambda_h \dot{e}_h + \ddot{e}_h \quad (\text{IV.34})$$

IV.4.2 Calcul de la commande équivalente

Calculons d'abord \dot{S}

$$\dot{S} = \lambda^2 \dot{e} + 2\lambda \ddot{e} - \ddot{x}_{1d} + \ddot{x}_1 \quad (\text{IV.35})$$

Remplaçons la valeur de \ddot{x}_1 dans (IV.35)

- On aura Pour Σ_v :

$$\dot{S}_v = \lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \ddot{x}_{1d} + a_v \left[\frac{\partial f_v(x_3)}{\partial x_3} \dot{x}_3 - \frac{\partial g_v(x_1)}{\partial x_1} \dot{x}_1 - b_v \dot{x}_2 \right] \quad (\text{IV.36})$$

Remplaçons \dot{x}_3 et \dot{x}_1 par leurs valeurs dans (IV.36), on trouve

$$\dot{S}_v = \lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \ddot{x}_{1d} - a_v c_v \frac{\partial f_v(x_3)}{\partial x_3} x_3 - a_v d_v \frac{\partial f_v(x_3)}{\partial x_3} u_v - a_v^2 \frac{\partial g_v(x_1)}{\partial x_1} x_2 - a_v b_v \dot{x}_2 \quad (IV.37)$$

Sur la surface de glissement, on a $\dot{S}_v = 0$, par conséquent la commande équivalente est :

$$u_{v_{eq}} = \frac{-1}{a_v d_v \frac{\partial f_v(x_3)}{\partial x_3}} \left[\lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \ddot{x}_{1d} - a_v c_v \frac{\partial f_v(x_3)}{\partial x_3} x_3 - a_v^2 \frac{\partial g_v(x_1)}{\partial x_1} x_2 - a_v b_v \dot{x}_2 \right] \quad (IV.38)$$

- Pour Σ_h on a :

$$\dot{S}_h = \lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \ddot{x}_{1d} + a_h \left[\frac{\partial f_h(x_3)}{\partial x_3} \dot{x}_3 - b_h \dot{x}_2 \right] \quad (IV.39)$$

De même que précédemment on aura :

$$\dot{S}_h = \lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \ddot{x}_{1d} - a_h c_h \frac{\partial f_h(x_3)}{\partial x_3} x_3 - a_h d_h \frac{\partial f_h(x_3)}{\partial x_3} u_h - a_h b_h \dot{x}_2 \quad (IV.40)$$

Et la commande équivalente est :

$$u_{h_{eq}} = \frac{-1}{a_h d_h \frac{\partial f_h(x_3)}{\partial x_3}} \left[\lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \ddot{x}_{1d} - a_h c_h \frac{\partial f_h(x_3)}{\partial x_3} x_3 - a_h b_h \dot{x}_2 \right] \quad (IV.41)$$

IV.4.3 L'attractivité

La condition de convergence est :

$$\dot{S}(x)S(x) < 0 \quad (IV.42)$$

Soit la commande

$$u_\eta = -K_u \text{signe}(S) \quad (IV.43)$$

La loi de commande globale est alors :

$$u = u_{eq} - K_u \text{signe}(S) \quad (IV.44)$$

Injectons u dans l'équation (IV.35)

$$\dot{S} = -K_u \text{signe}(S) \quad (\text{IV.45})$$

Multiplions (IV.45) par S

$$\dot{S}(x)S(x) = -S(x)K_u \text{signe}(S(x)) < 0 \quad (\text{IV.46})$$

Ce qui vérifie la condition de convergence (IV.42)

En somme, Les lois de commandes finales sont :

- Pour Σ_v

$$u_v = \frac{-1}{a_v d_v \frac{\partial f_v(x_3)}{\partial x_3}} \left[\lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \ddot{x}_{1d} - a_v c_v \frac{\partial f_v(x_3)}{\partial x_3} x_3 - a_v^2 \frac{\partial g(x_1)}{\partial x_1} x_2 - a_v b_v \dot{x}_2 \right] - K_{uv} \text{signe}(S_v) \quad (\text{IV.47})$$

- Pour Σ_h

$$u_h = \frac{-1}{a_h d_h \frac{\partial f_h(x_3)}{\partial x_3}} \left[\lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \ddot{x}_{1d} - a_h c_h \frac{\partial f_h(x_3)}{\partial x_3} x_3 - a_h b_h \dot{x}_2 \right] - K_{uh} \text{signe}(S_h) \quad (\text{IV.48})$$

IV.4.4 Etude de la dynamique réduite

L'étude est la même pour les deux systèmes, horizontal et vertical

En régime glissant on a :

$$\begin{cases} S = 0 \\ \dot{S} = 0 \end{cases} \quad (\text{IV.49})$$

Et aussi

$$u = u_{eq} \quad (\text{IV.50})$$

On prend comme vecteur d'état

$$\begin{cases} e_1 = \dot{e} \\ e_2 = \ddot{e} \\ e_3 = \dddot{e} \end{cases} \quad (\text{IV.51})$$

Avec

$$e = x_1 - x_{1d} \quad (\text{IV.52})$$

Le système Σ devient alors

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = e_3 \\ \dot{e}_3 = \ddot{e} \end{cases} \quad (IV.53)$$

Or d'après (IV.49) on a :

$$\ddot{e} = \lambda^2 \dot{e} + 2\lambda \ddot{e} \quad (IV.54)$$

Injectons (IV.54) dans (IV.53), on trouve

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = e_3 \\ \dot{e}_3 = \lambda^2 \dot{e} + 2\lambda \ddot{e} \end{cases} \quad (IV.55)$$

Les valeurs propres du système (IV.55) sont

$$VP(\Sigma) = \begin{bmatrix} -\lambda \\ -\lambda \\ 0 \end{bmatrix} \quad (IV.56)$$

Comme λ est positif, on conclut que le système est L-stable.

Or, Lorsque $S = 0$ on a

$$e = e_1 = 0 \quad (IV.57)$$

D'après (IV.55) et (IV.57) :

$$\begin{cases} e_2 \rightarrow 0 \\ e_3 \rightarrow 0 \end{cases} \quad (IV.58)$$

Par conséquent, la dynamique réduite sur la surface de glissement est **stable**

IV.4.5 Simulation

Les figures (IV.5) et (IV.6) représentent les résultats obtenus avec une fonction d'attractivité $(-k \cdot \text{signe}(s))$, cette dernière est à l'origine du phénomène de chattering, qui apparaît au niveau des commandes et des vitesses sous forme d'oscillations à fréquence infinie.

L'utilisation d'une fonction d'attractivité plus lisse (sigmoïde), a permis l'élimination du chattering et l'adoucissement de la commande, qui est devenue admissible par les actionneurs, tout en conservant les mêmes performances. figures (IV.7) et (IV.8)

Enfin, la figure (IV.9) montre les résultats de l'application des régulateurs SISO sur le modèle couplé avec une fonction d'attractivité sigmoïde. On constate un rejet des effets de couplages.

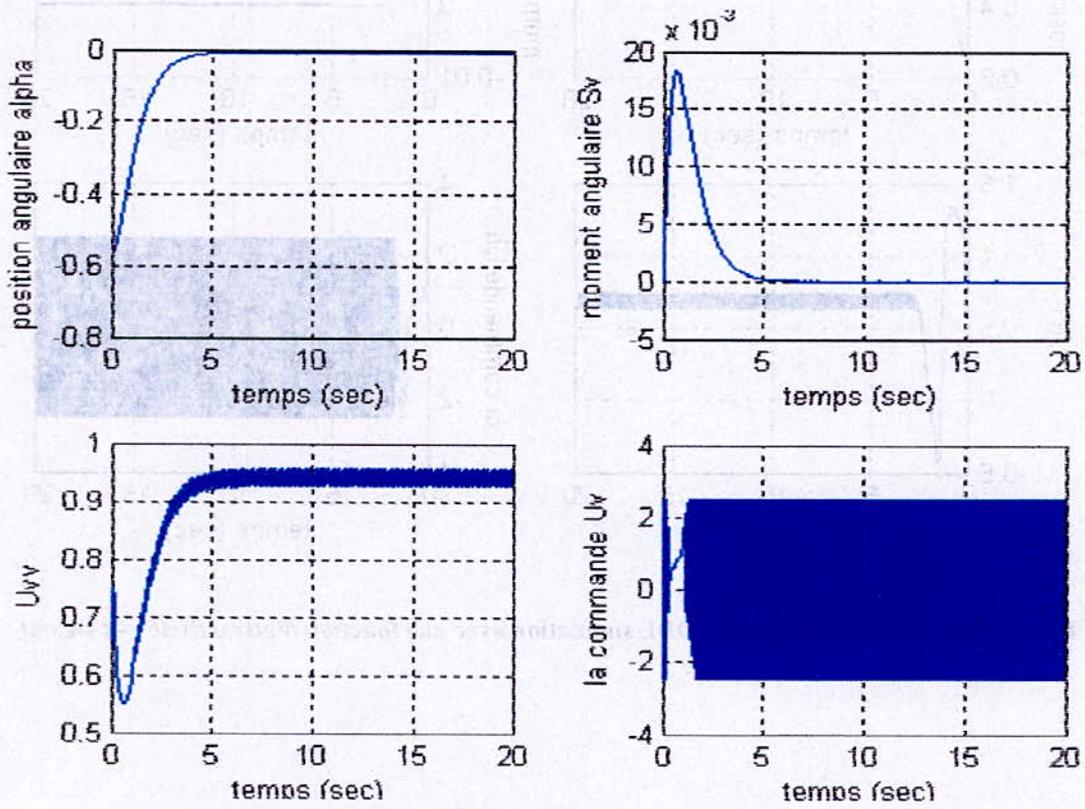


Figure IV.5 : modèle vertical IDDL simulation avec une fonction d'attractivité : $-k \text{ sign}(s)$



Figure IV.5 : modèle vertical IDDL simulation avec une fonction d'attractivité : $-k \text{ sign}(s)$

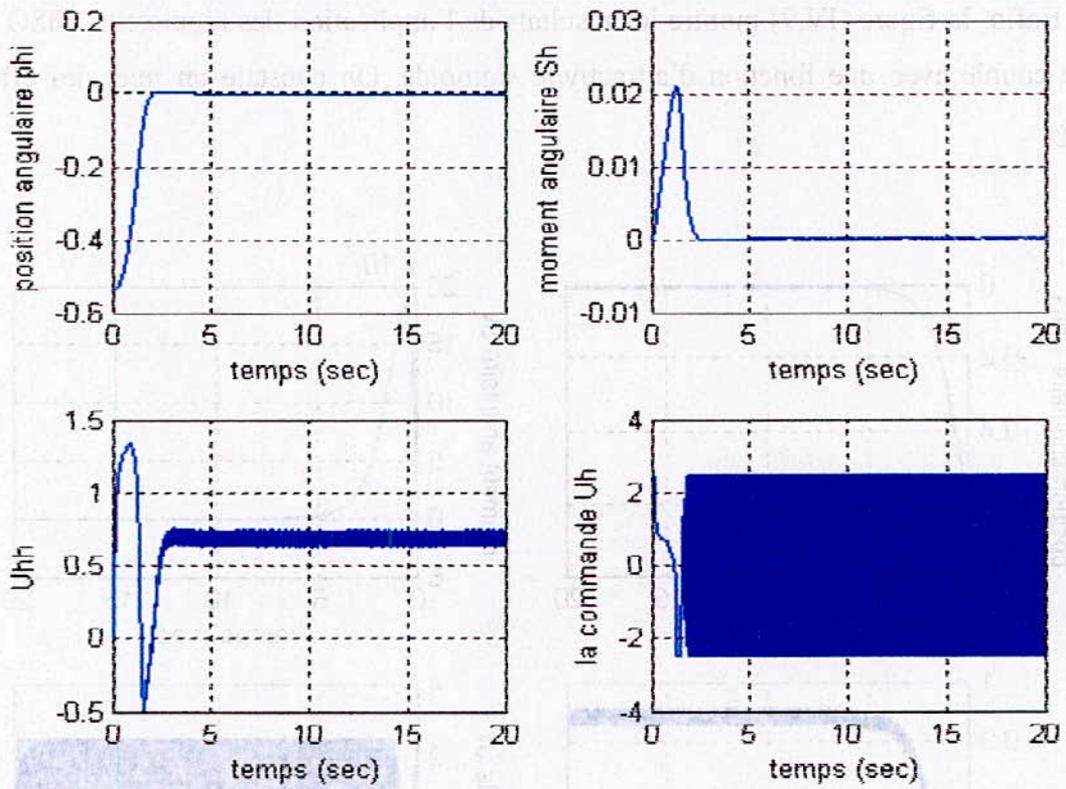


Figure IV.6 : modèle horizontal 1DDL simulation avec une fonction d'attractivité : $-k \text{sign}(s)$

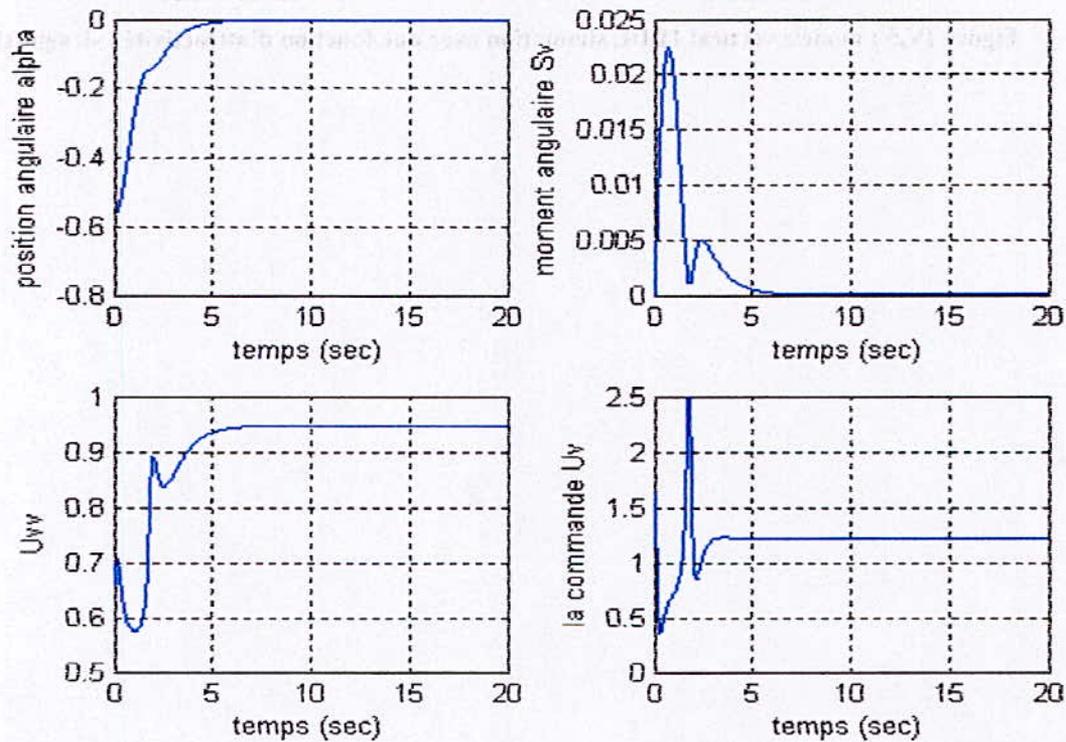


Figure IV.7 : modèle vertical 1DDL simulation avec une fonction d'attractivité : $-k \text{sigmoïde}(s)$ ($\lambda = 5$)

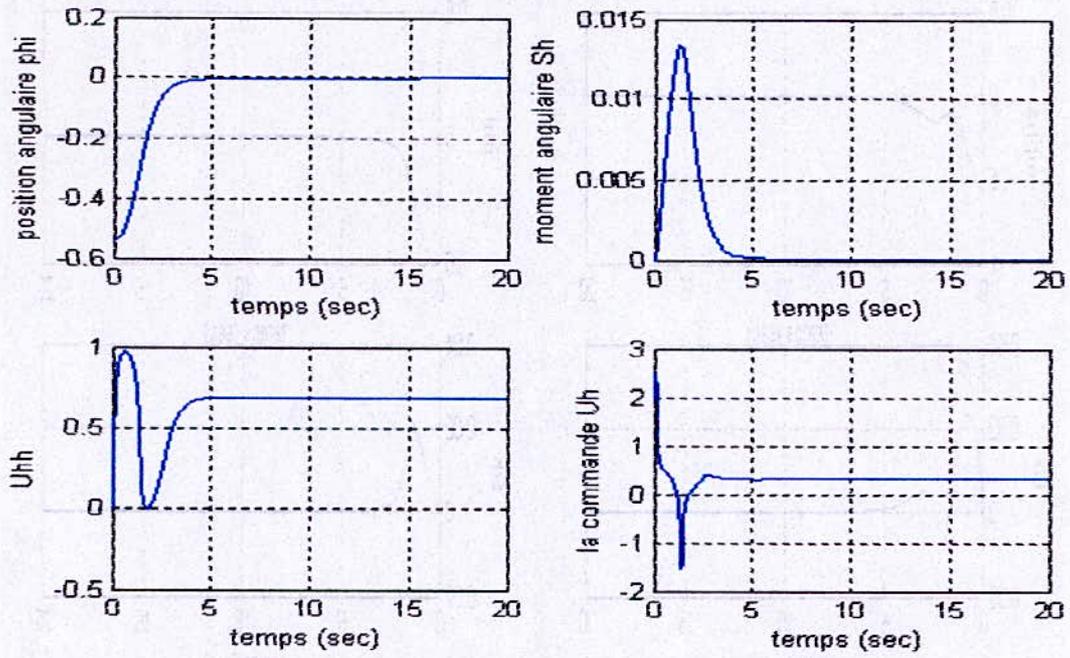


Figure IV.8 : modèle horizontal 1DDL simulation avec une fonction d'attractivité : $-k \text{ sigmoïde}(s)$ ($\lambda = 5$)

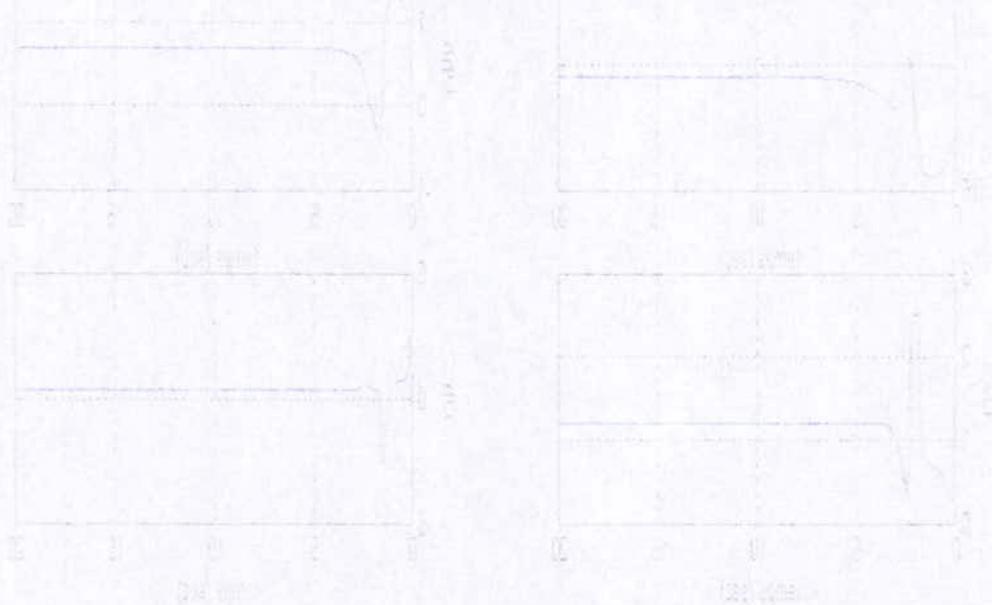


Figure IV.8 : modèle horizontal 1DDL simulation avec une fonction d'attractivité : $-k \text{ sigmoïde}(s)$ ($\lambda = 5$)

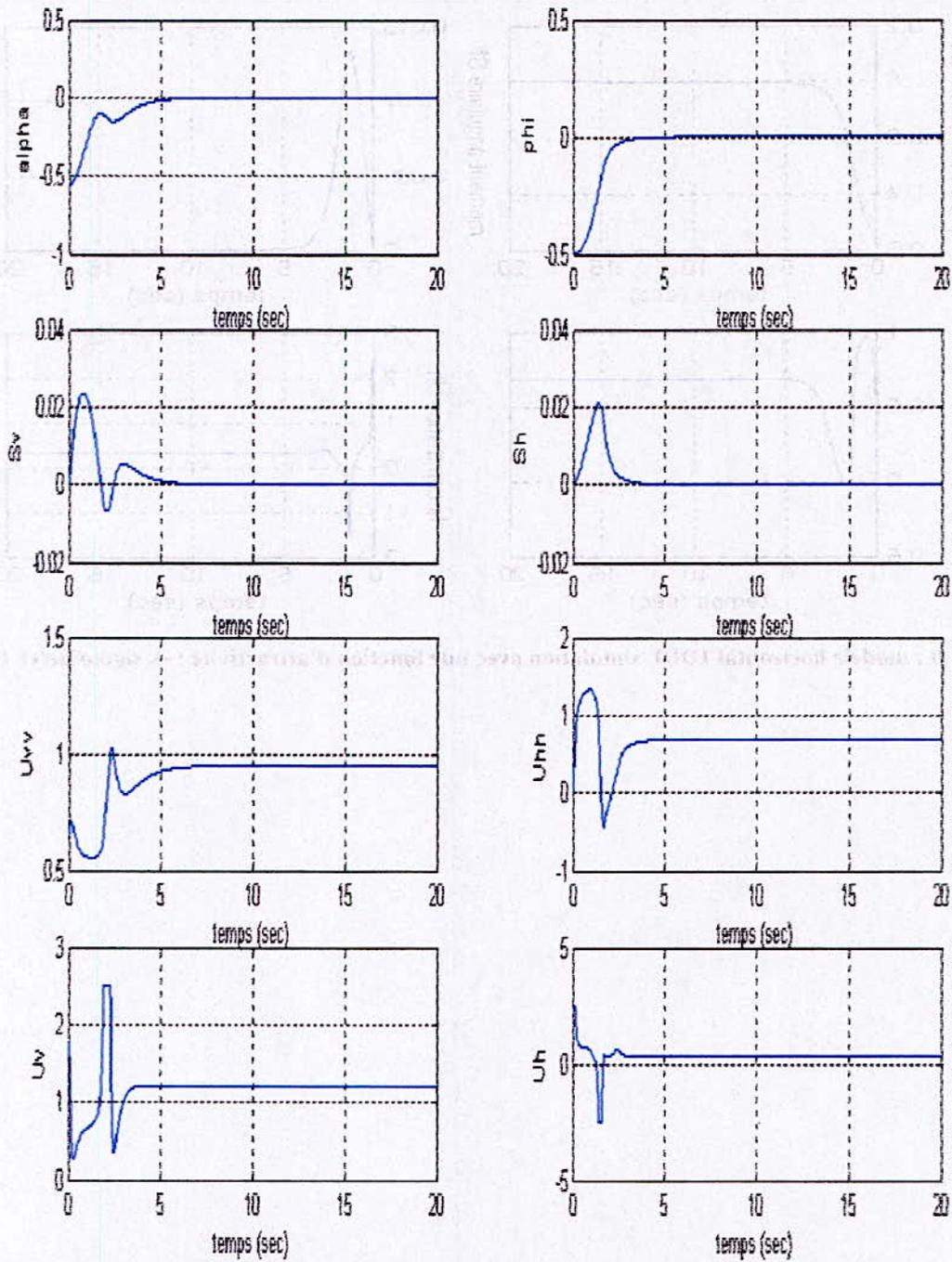


Figure IV.9 : modèle couplé 2DDL simulation avec une fonction d'attractivité : $-k$ sigmoïde(s) ($\lambda = 5$)

IV.5 REGULATEUR FLOU GLISSANT

L'inconvénient majeur de la commande par mode de glissement est le chattering au niveau de la commande. L'approximation de la fonction discontinue d'attractivité par une fonction continue est l'une des méthodes pour remédier à ce problème. Nous proposons ici une technique alternative, qui consiste à exploiter la logique floue pour assurer l'attractivité

IV.5.1 Structure du contrôleur flou glissant [lin94][liao94]

Le contrôleur flou glissant est composé de deux blocs (figure IV.10). Un bloc pour le calcul de la commande équivalente et la surface de glissement. Le deuxième bloc est un régulateur flou, dont les entrées sont la surface et sa variation (variation de surface), et la commande attractante comme sortie. La sortie du contrôleur flou glissant est la somme de la commande équivalente et la commande attractante

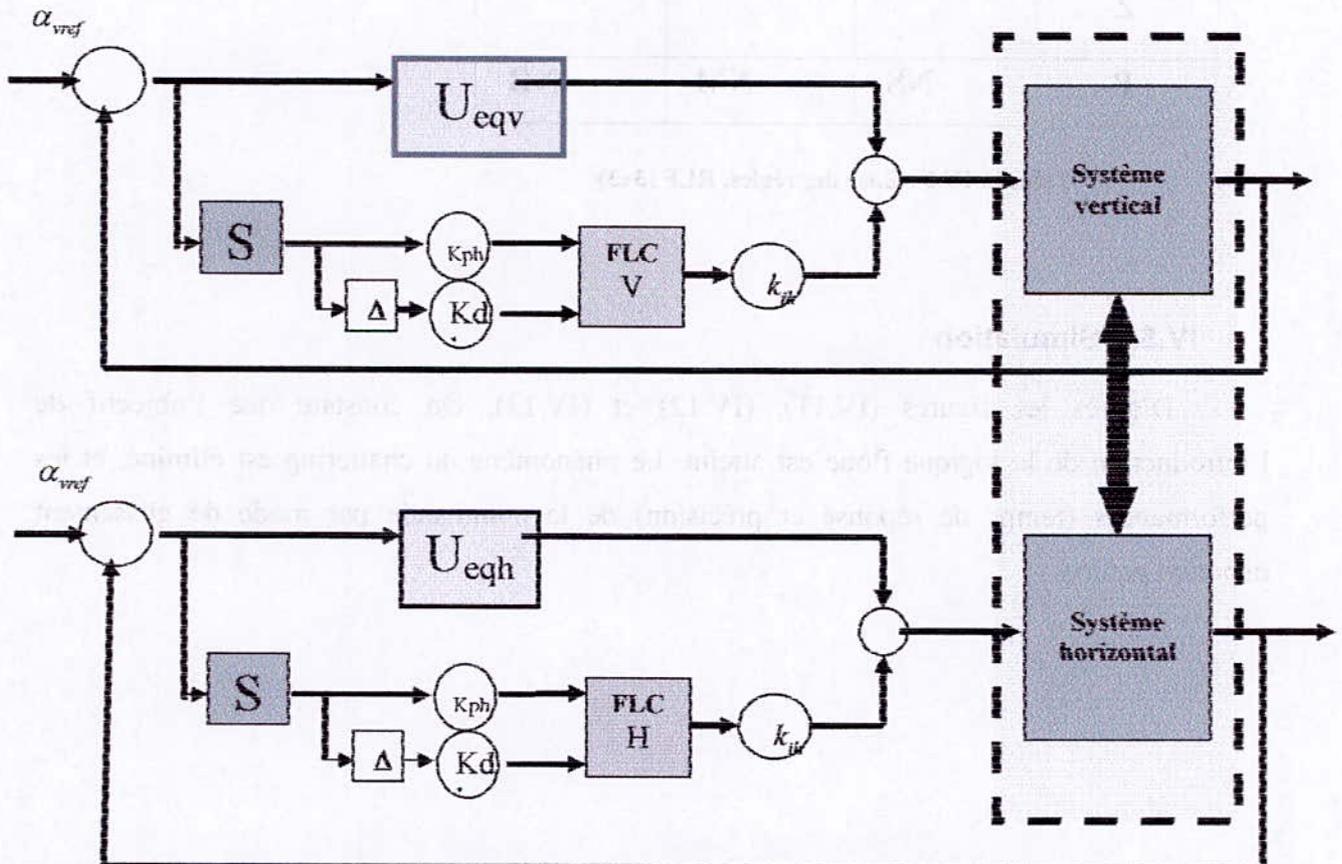


Figure IV.10 structure du régulateur flou glissant

IV.5.2 Synthèse du contrôleur flou glissant

IV.5.2.1 Commande équivalente et surface de glissement

Le calcul de la commande équivalente et de la surface de glissement est le même que pour la commande par mode de glissement développée précédemment.

IV.5.2.2 Commande d'attractivité

L'attractivité est réalisée par un régulateur flou. Les entrées (surface et sa variation) ont trois fonctions d'appartenance, quant à la sortie (la commande d'attractivité), elle en possède cinq. La base des règles est représentée dans le tableau suivant :

s \ Δs	N	Z	P
N	PB	PM	PS
Z	PS	ZE	NS
P	NS	NM	NB

Tableau IV.2 : Base des règles, RLF (3x3)

IV.5.3 Simulation

D'après les figures (IV.11), (IV.12) et (IV.13), On constate que l'objectif de l'introduction de la logique floue est atteint. Le phénomène du chattering est éliminé, et les performances (temps de réponse et précision) de la commande par mode de glissement demeure gardées.

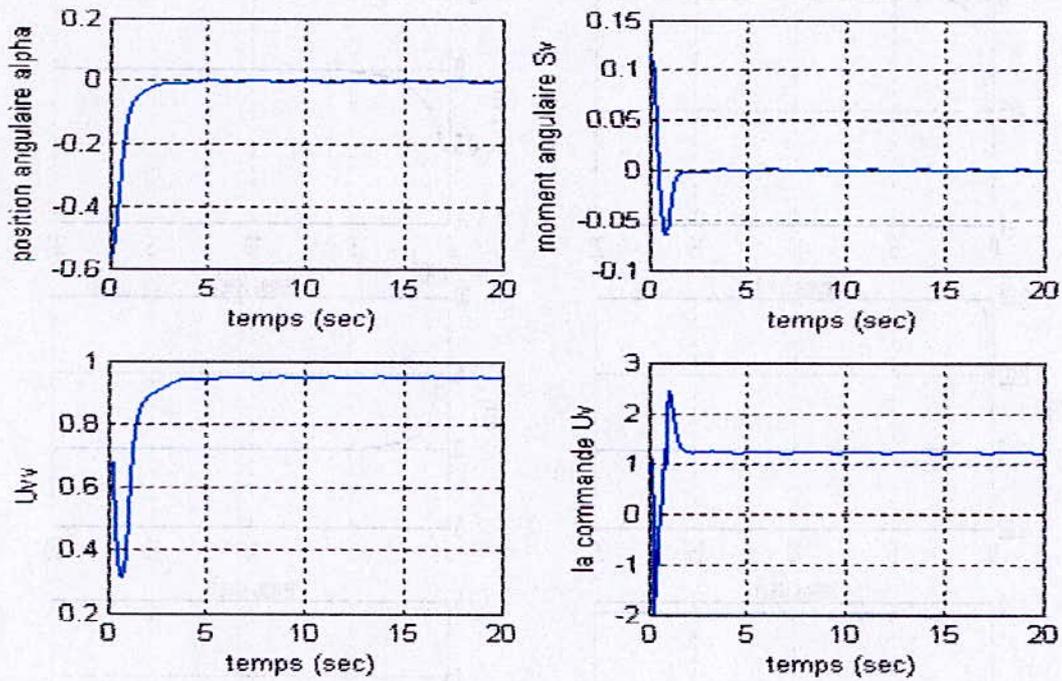


Figure IV.11 : modèle vertical IDDL

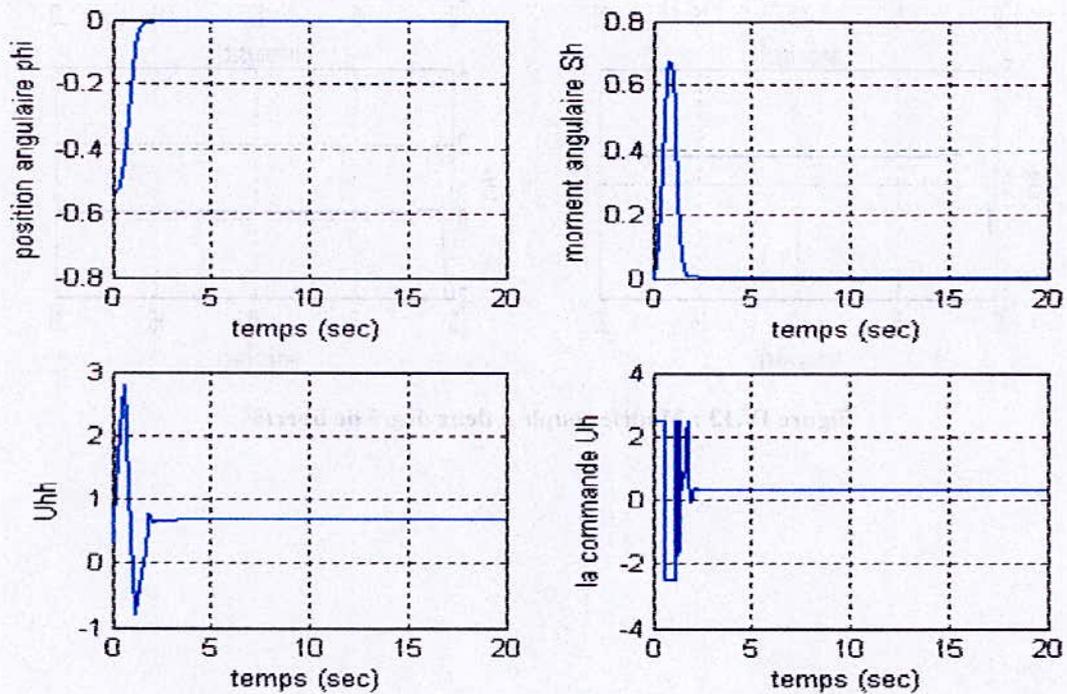


Figure IV.12 : modèle horizontal IDDL

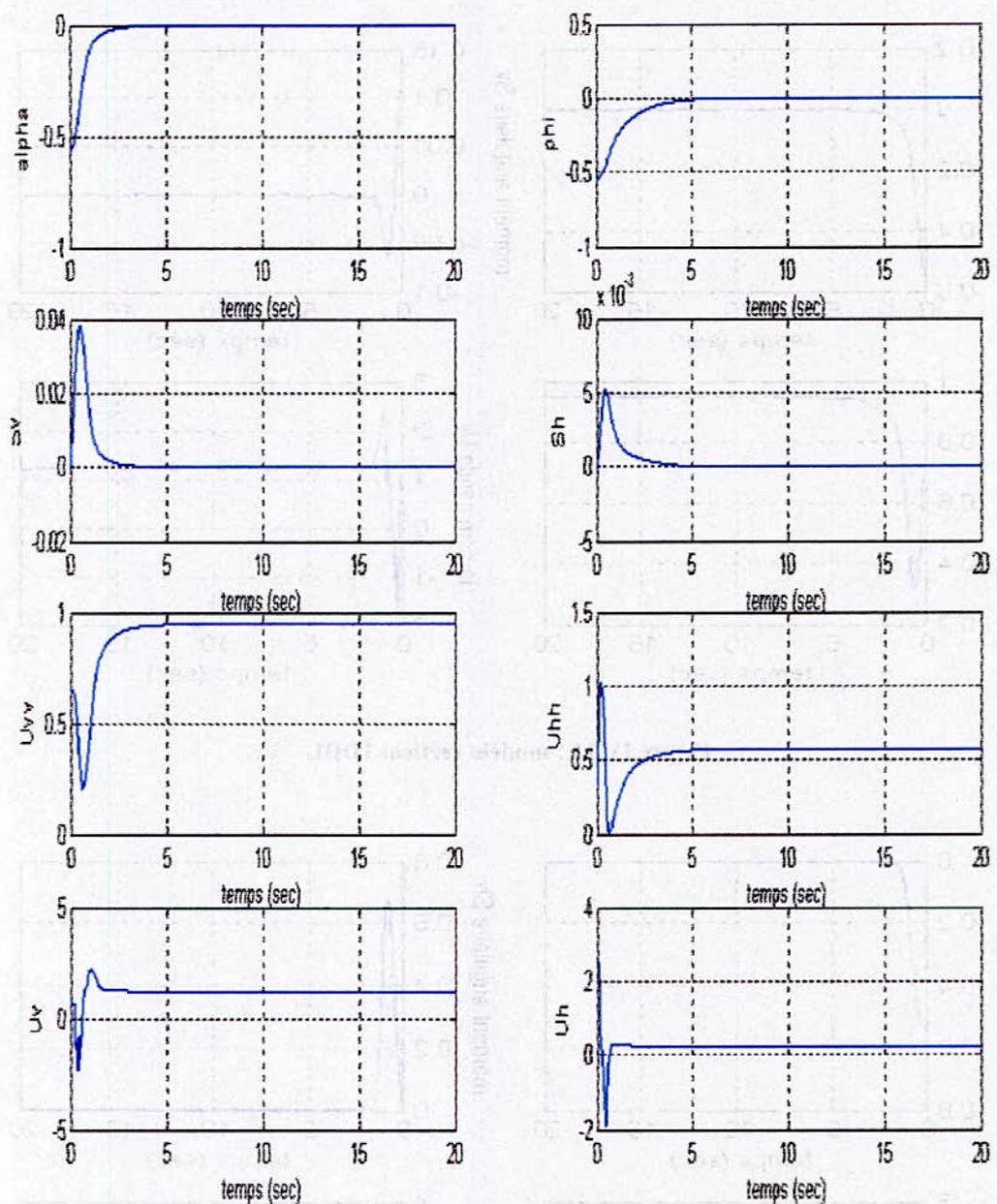


Figure IV.13 : Modèle couplé à deux degré de liberté

IV.6 CONCLUSION

Dans ce chapitre nous nous sommes intéressés, à l'application des commandes floues, bouclage non linéaire, commande par mode de glissement, et la commande floue glissante au simulateur d'hélicoptère. La commande floue ne demande pas beaucoup de développement théorique, mais sa difficulté se situe au niveau du choix heuristique des paramètres fastidieux. Par contre les autres techniques ont une méthode de synthèse systématique. Nous avons vu aussi qu'il est judicieux d'utiliser une commande décentralisée.

Les résultats de simulation montrent que les commandes synthétisées pour les modèles à un degré de liberté, ont pu rejeter l'effet de couplage lors de leur application au modèle à deux degrés de liberté. Nous avons vu aussi que le problème majeur de la commande par mode de glissement, qui est le chattering peut être éliminé, soit par l'utilisation de la fonction d'attractivité sigmoïde, ou par l'utilisation de la logique floue pour adoucir la commande.

CHAPITRE V

***IMPLEMENTATION ET RESULTATS
EXPERIMENTAUX***

V IMPLEMENTATION ET RESULTATS EXPERIMENTAUX

V.1 INTRODUCTION

Dans Cette partie, nous présentons les résultats obtenus de l'application en temps réel des lois de commandes synthétisées précédemment, au simulateur d'hélicoptère

Ces lois de commandes ont été testées d'abord pour des problèmes de stabilisation et de poursuite de trajectoires. En suite des variations paramétriques ainsi que des perturbations extérieures ont été introduites pour évaluer la robustesse.

La variation paramétrique consistait à changer la position du contrepoids, quant aux perturbations externes, une charge de 0.12 N.m (250g) était accrochée à la poutre après sa stabilisation

V.2 CONSIDERATIONS PRATIQUES

Plusieurs points doivent être considérés avant l'implémentation d'une commande sur un système réel, et plus particulièrement sur le simulateur. Voici quelques une des considérations :

Comme Les commandes synthétisées sont continues, le choix de la fréquence d'échantillonnage a une grande influence sur les performances du régulateur. Une valeur élevée est nécessaire pour simuler le fonctionnement continu du régulateur figure (V.2). Cependant, il faut trouver un compromis entre temps de calcul et période d'échantillonnage figure (V.1)

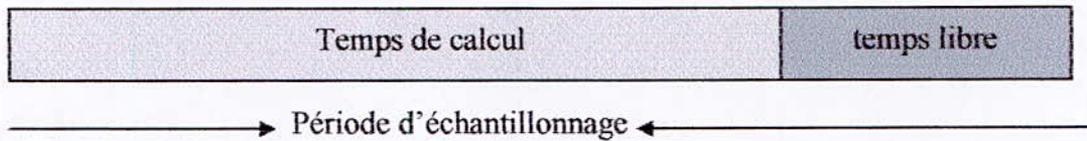


Figure V.1 : temps de calcul VS période d'échantillonnage

En somme, on doit avoir $T_{calcul} < T_{echantillonnage}$
 $T_{echantillonnage}$ petite \Rightarrow travailler en pseudo continu

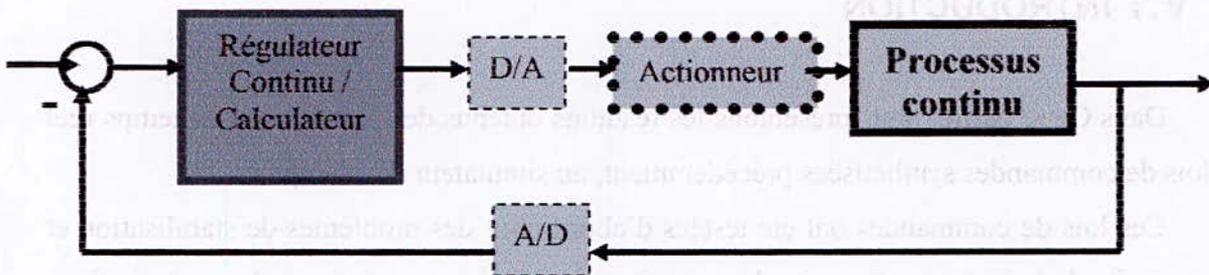
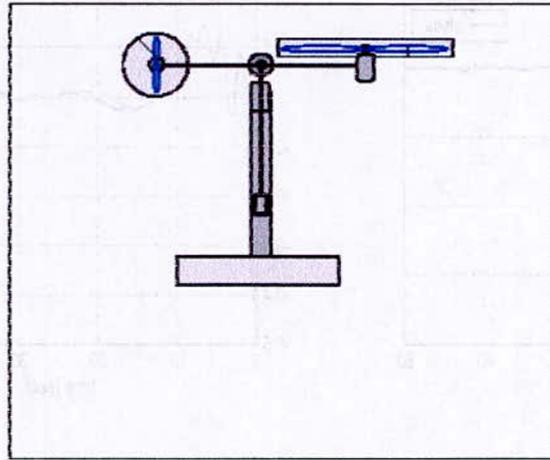


Figure V.2 : Structure de la boucle de régulation

A fin de réduire l'effet des bruits de mesures, nous avons placé en aval de chaque sortie de capteur utilisée par le régulateur un filtre passe-bas, ce type de filtre est implémenté comme étant un système de premier ordre, il atténue les signaux avec des fréquences supérieures à la fréquence de coupure. Cependant, cette fréquence de coupure ne doit pas être si petite au risque d'atténuer aussi le signal de mesure

Dans la partie synthèse, nous avons supposé qu'on a un retour complet du vecteur d'état. Or dans l'implémentation, on récupère seulement la position de la poutre (c'est-à-dire l'angle d'élévation et l'angle d'azimut), pour reconstruire les états non mesurables, nous avons ajouté un bloc qui estime ces états à partir du modèle mathématique,

Lors des expériences, il faut veiller à initialiser les capteurs sur la position illustrée dans la figure (V.3) car elle représente l'origine de l'espace d'état, et cela avant l'exécution du programme de commande



Figure(V.3) : position (0,0)

V.3 LA COMMANDE PAR LOGIQUE FLOUE

La figure (V.4) montre que la stabilité est réalisée pour l'élévation comme pour l'azimut, mais avec un dépassement de 20% et une réponse relativement lente.

Quant à la poursuite, (références sinusoïdales de 0.02, 0.04 Hz et signal carré), elle est aussi réalisée comme illustré dans les figures (V.5, V.6 et V.7), mais avec une erreur de poursuite de plus en plus importante à mesure que la fréquence de la référence augmente. Ce-ci est dû au manque d'expertise dans le choix des fonctions d'appartenance, gains de normalisation et aussi les règles d'inférences.

La tension de commande reste dans les limites tolérées par les actionneurs, bien qu'elle présente des pics qui seront filtrés intrinsèquement par les moteurs (filtre passe bas).

En ce qui concerne la robustesse, on constate que le régulateur flou est robuste vis-à-vis des variations paramétriques (figure (V.8), figure (V.9)), comme pour le rejet des perturbations extérieurs (figure(V.10)), étant donné que le calcul de la commande n'est pas basée sur le modèle.

V.3.1 Stabilisation et poursuite

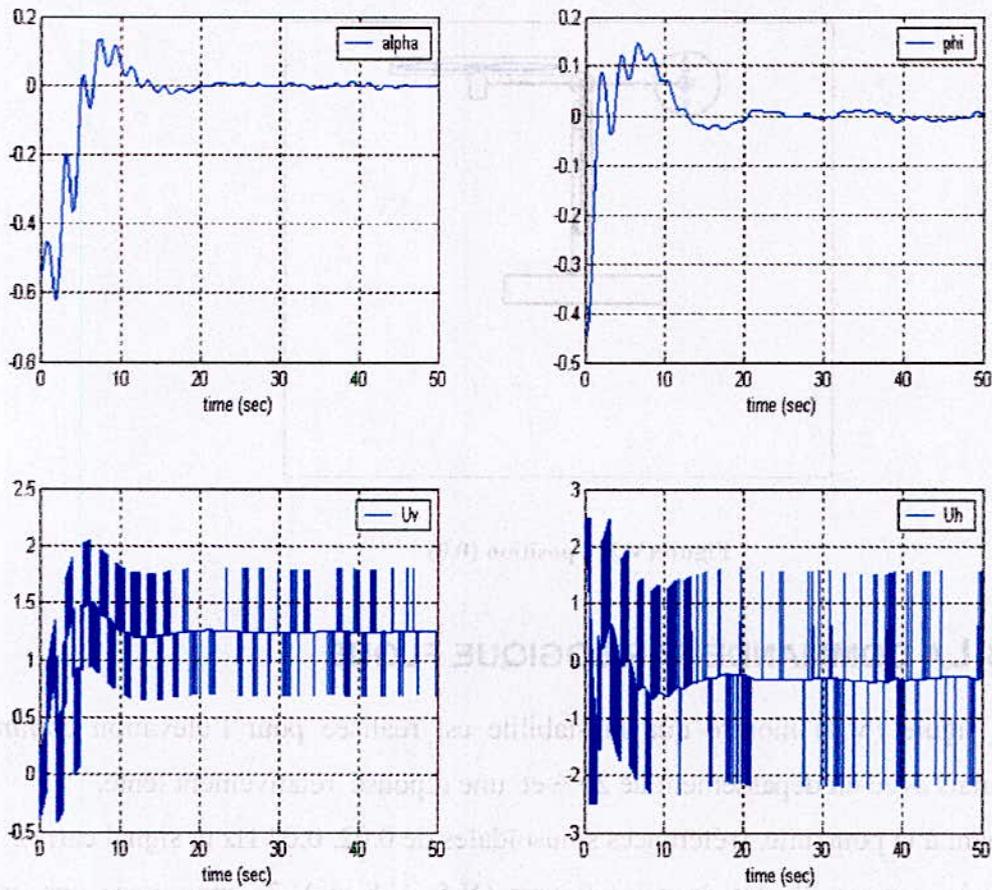


Figure V.4 : stabilisation du modèle à deux degré de liberté

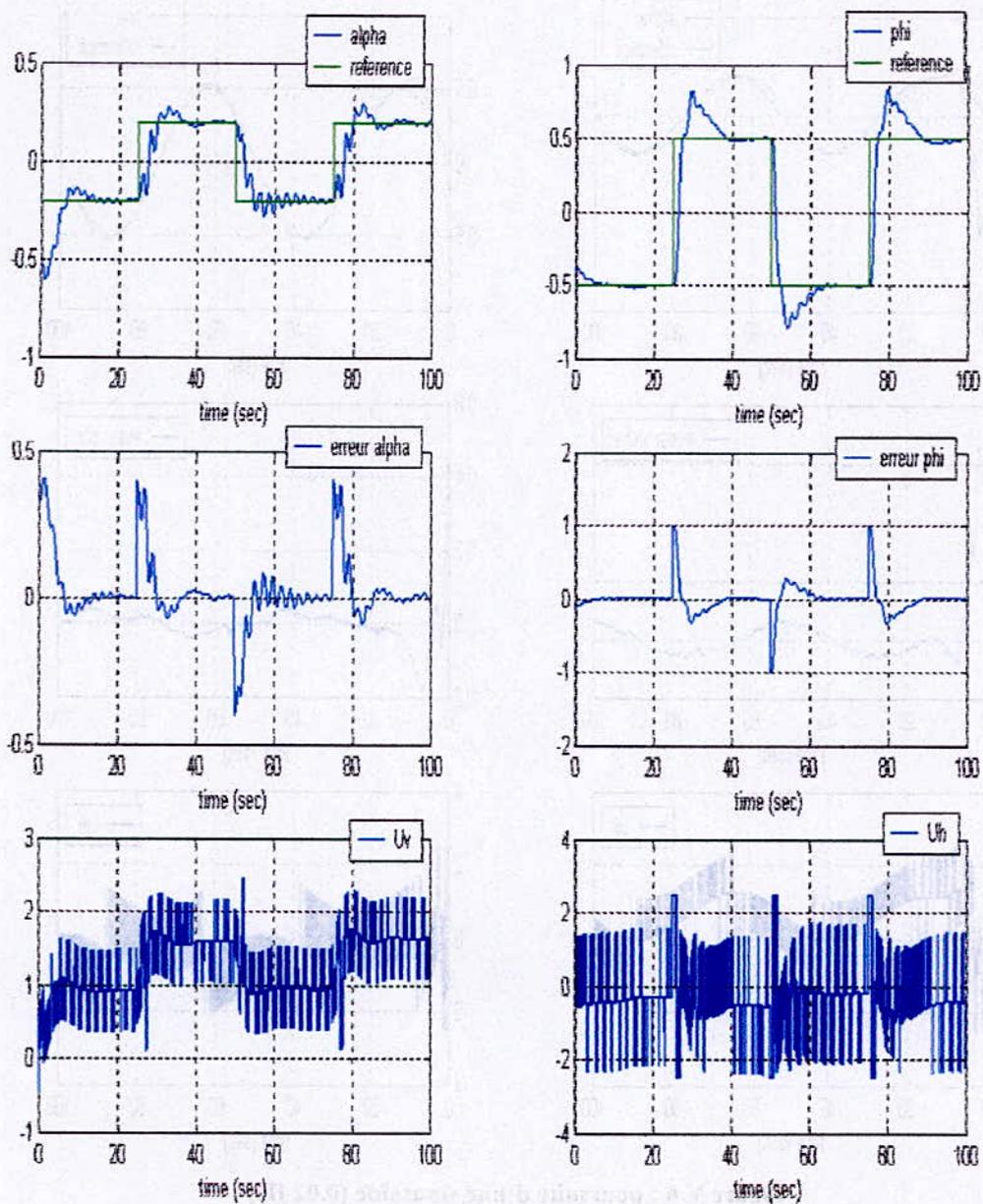


Figure V.5 : poursuite d'un signal carrée

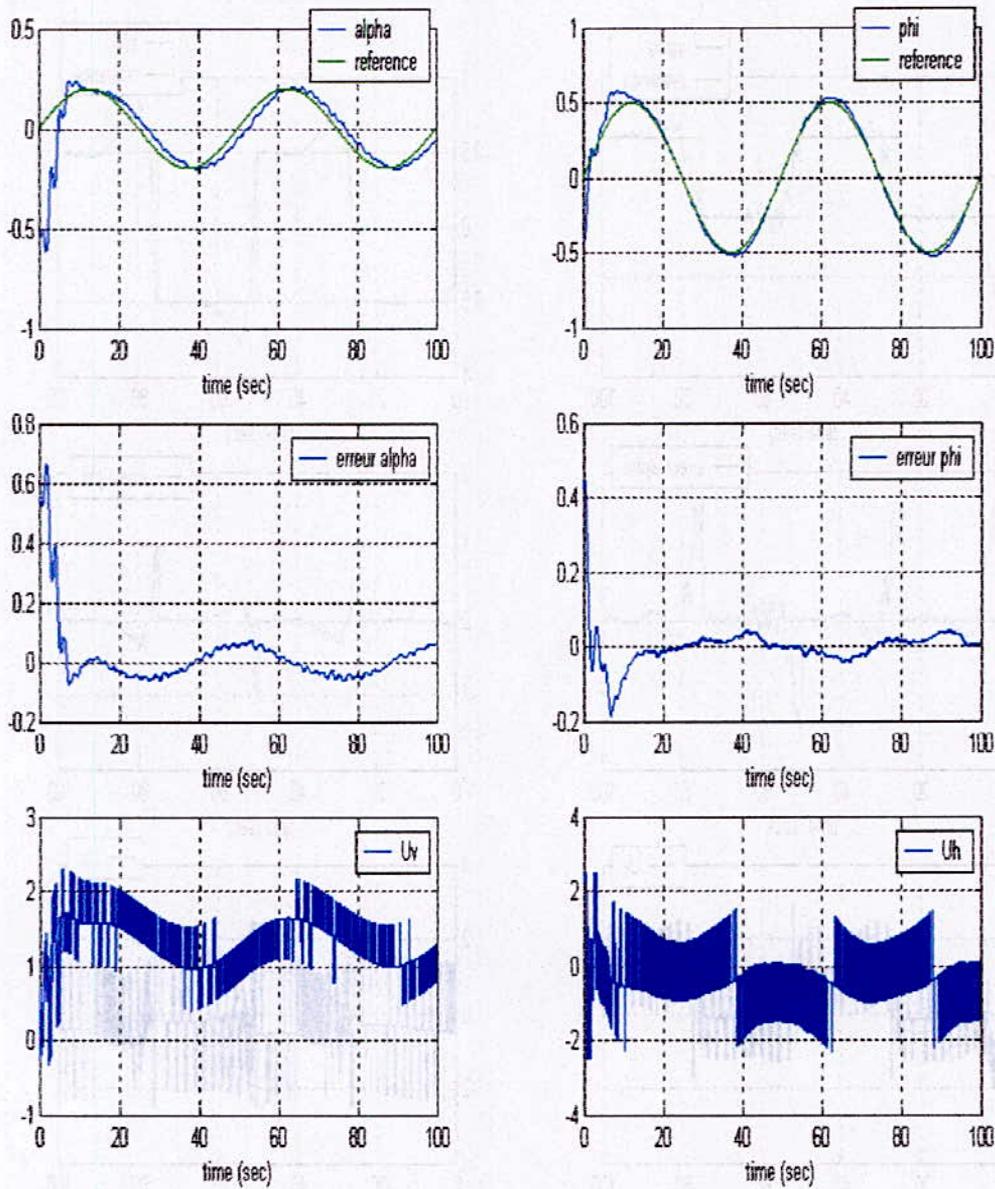


Figure V.6 : poursuite d'une sinusoïde (0.02 Hz)

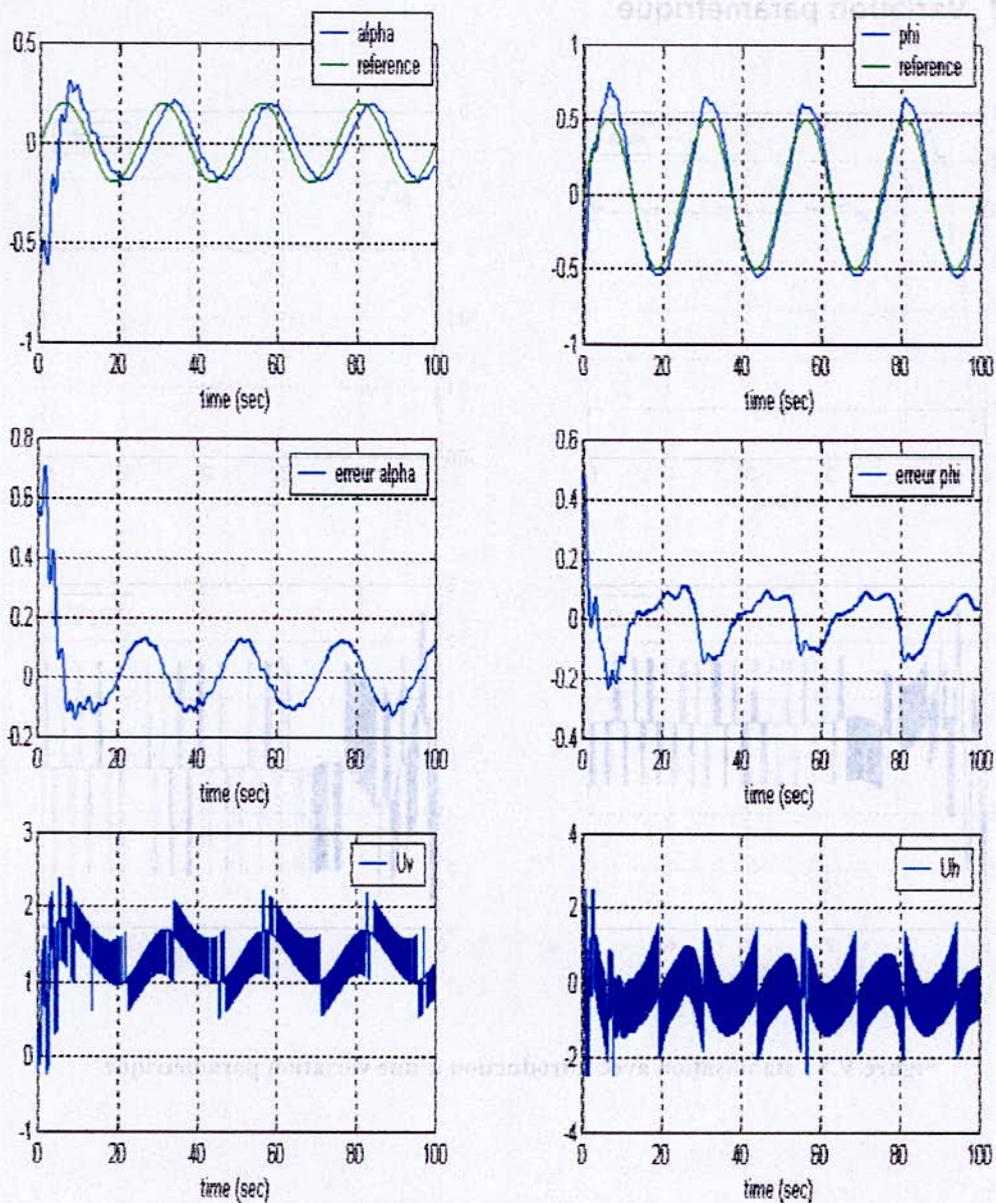


Figure V.7 : poursuite d'une sinusoïde (0.04 Hz)

V.3.2 Variation paramétrique

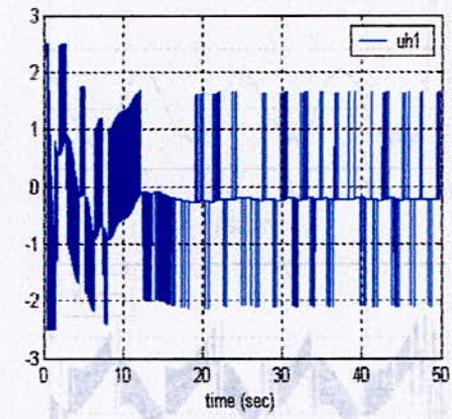
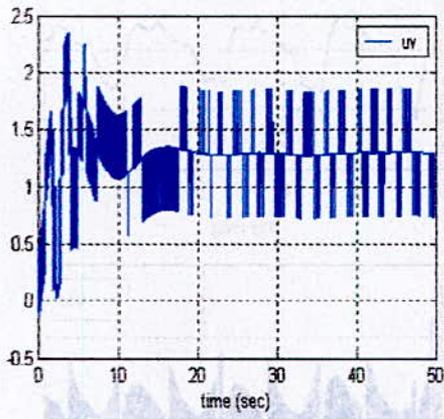
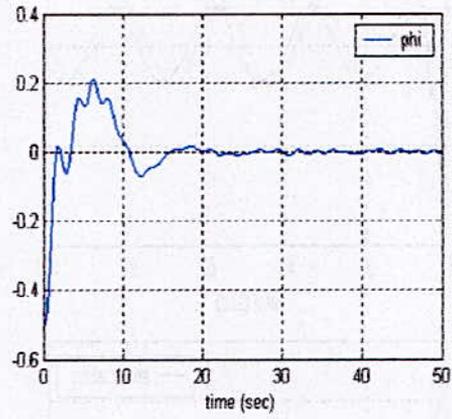
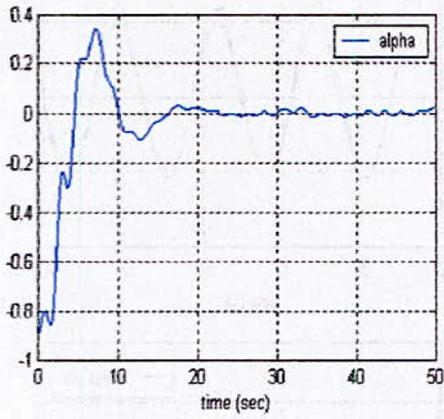


Figure V.8 : stabilisation avec introduction d'une variation paramétrique

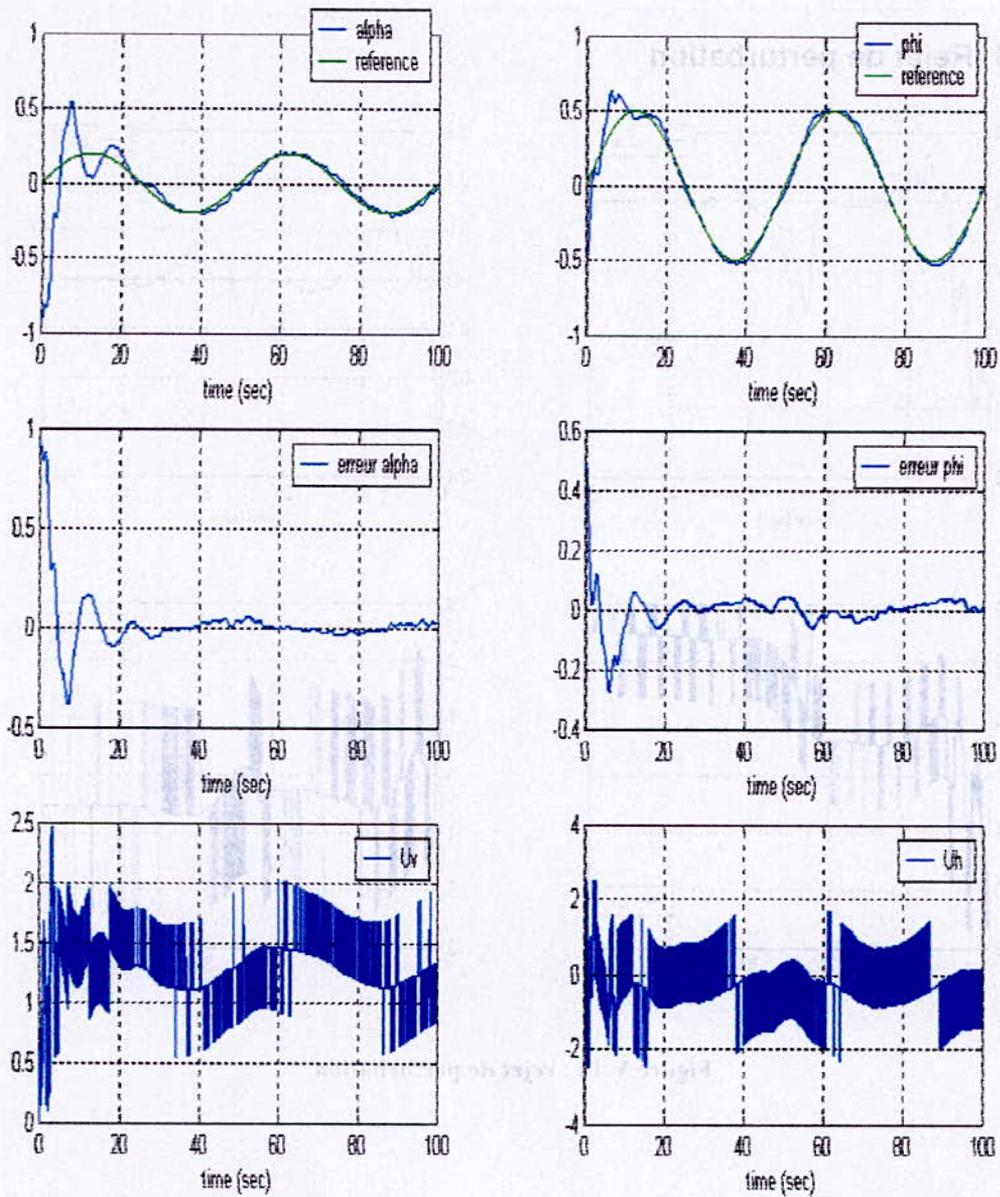


Figure V.9 : poursuite avec introduction d'une variation paramétrique

V.3.3 Rejet de perturbation

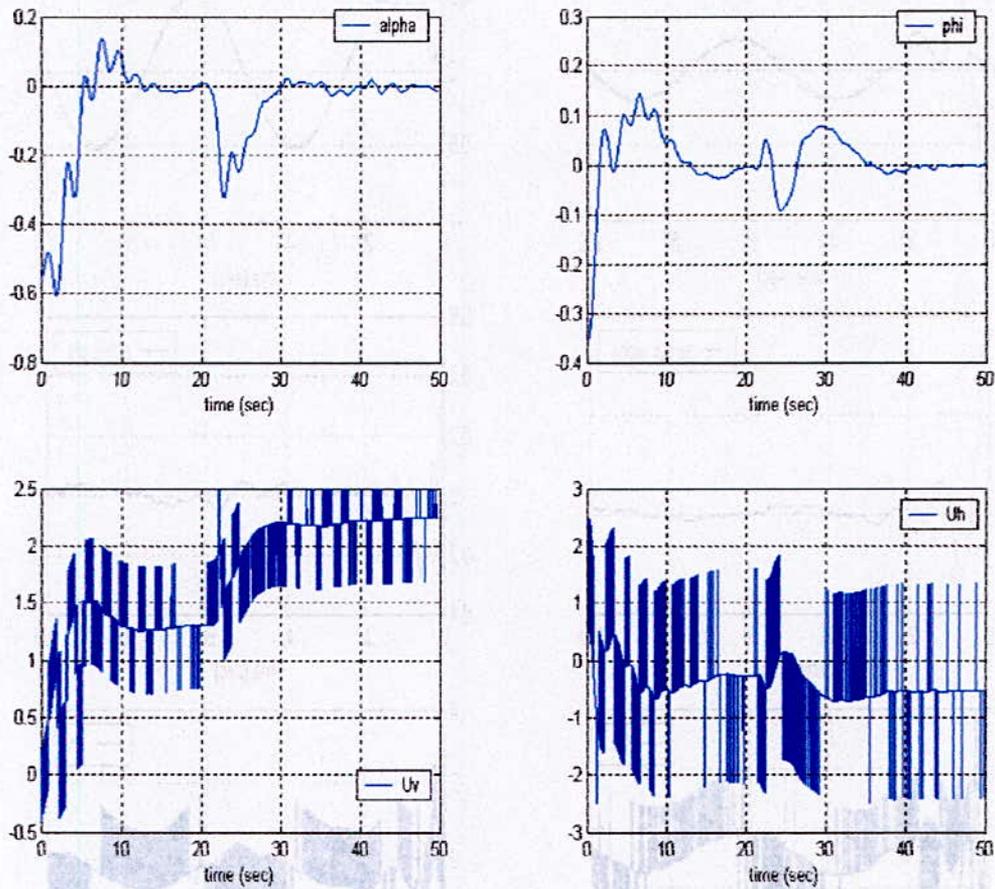


Figure V.10 : rejet de perturbation

Les performances non satisfaisantes obtenues avec le régulateur flou, nous ont amenés à synthétiser d'autres lois de commandes qui se basent sur le modèle mathématique du système, entre autre la présente commande (bouclage linéarisant).

V.4 COMMANDE PAR BOUCLAGE LINEARISANT

Les figures (V.11 et V.12) montrent les réponses de simulations avec celles du réel, on constate une légère différence entre ces réponses, ce qui est prévisible.

Les performances (temps de réponse sans dépassement) se sont améliorées tout en garantissant la stabilité. Car ces mêmes performances ont été imposées a priori, lors de la synthèse. Quant à la commande, elle est devenu douce et ne sollicite pas beaucoup les actionneurs. Figures (V.13, V.14, V.15, et V.16)

Cependant, cette commande n'est pas robuste vis-à-vis des perturbations extérieures comme il est indiqué dans la figure (V.18). Néanmoins, la robustesse vis-à-vis des variations paramétriques demeure relativement inchangées figure (V.17), du fait que ce paramètre a peu d'influence sur la loi de commande



V.4.1 Réponse réelle VS réponse du modèle

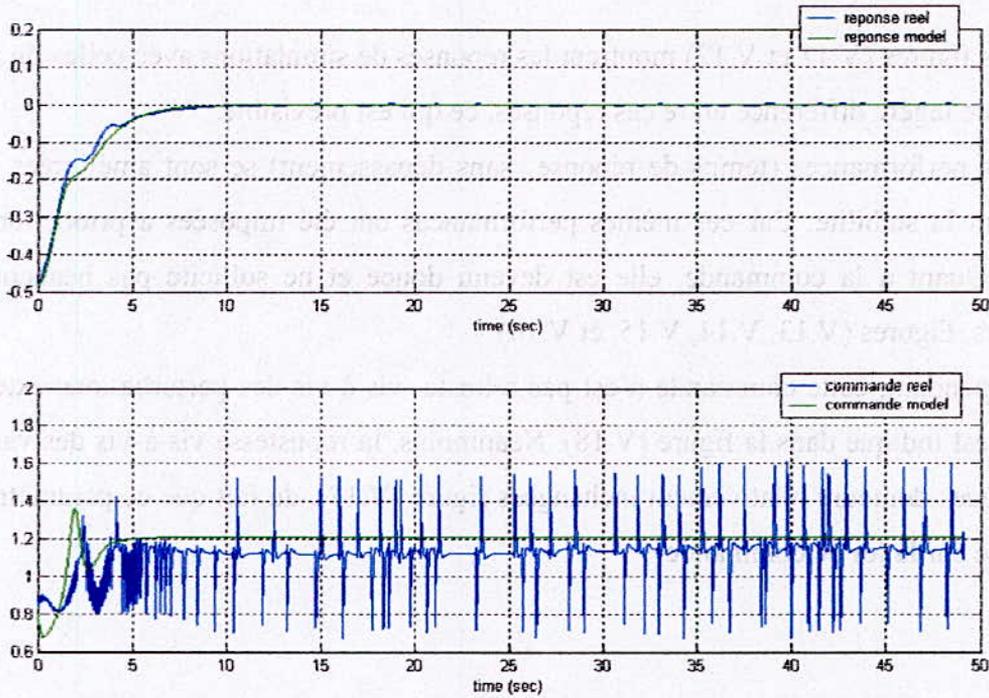


Figure V.11 : Réponse réelle VS réponse simulée (modèle vertical)

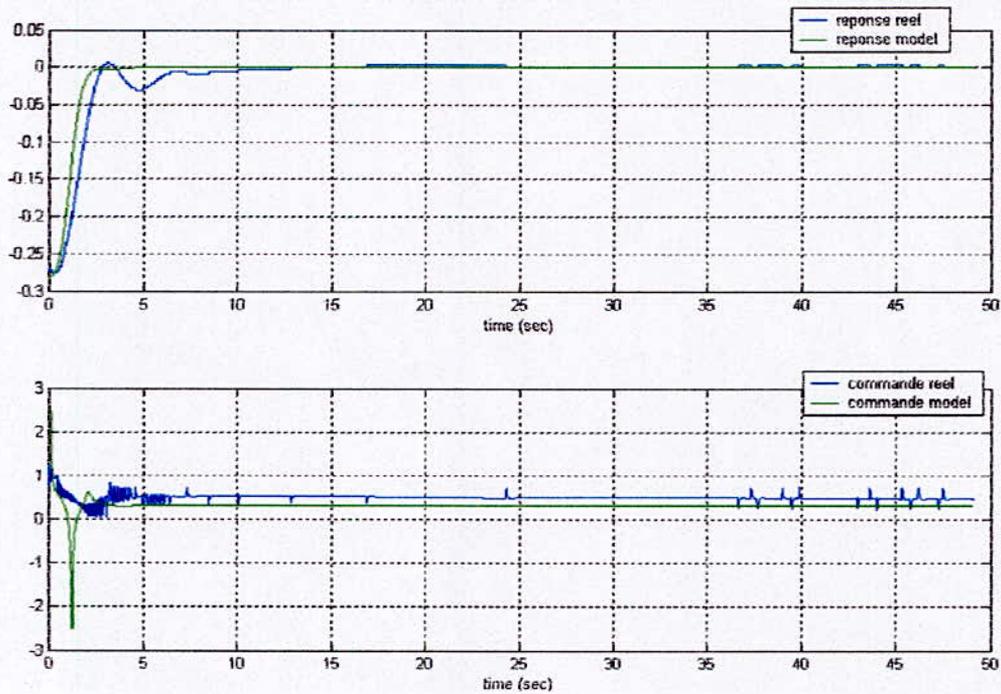


Figure V.12 : Réponse réelle VS réponse simulée (modèle horizontal)

V.4.2 Stabilisation et poursuite

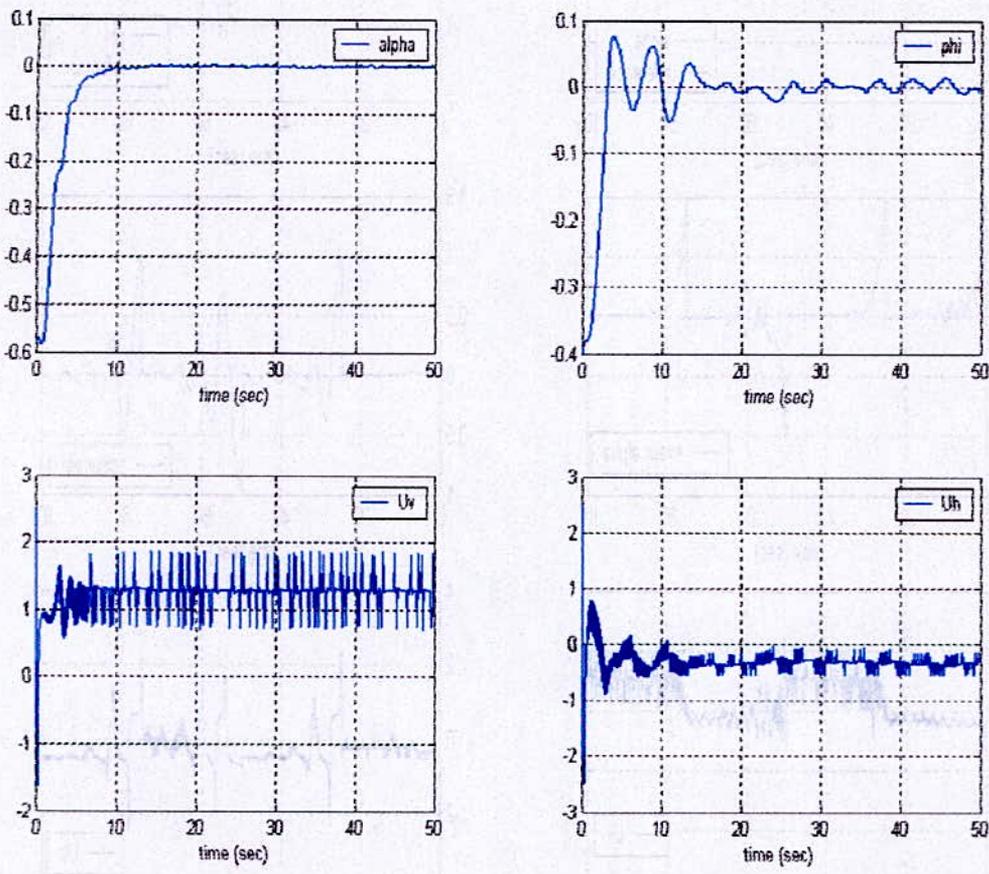


Figure V.13 : stabilisation du modèle à deux degré de liberté

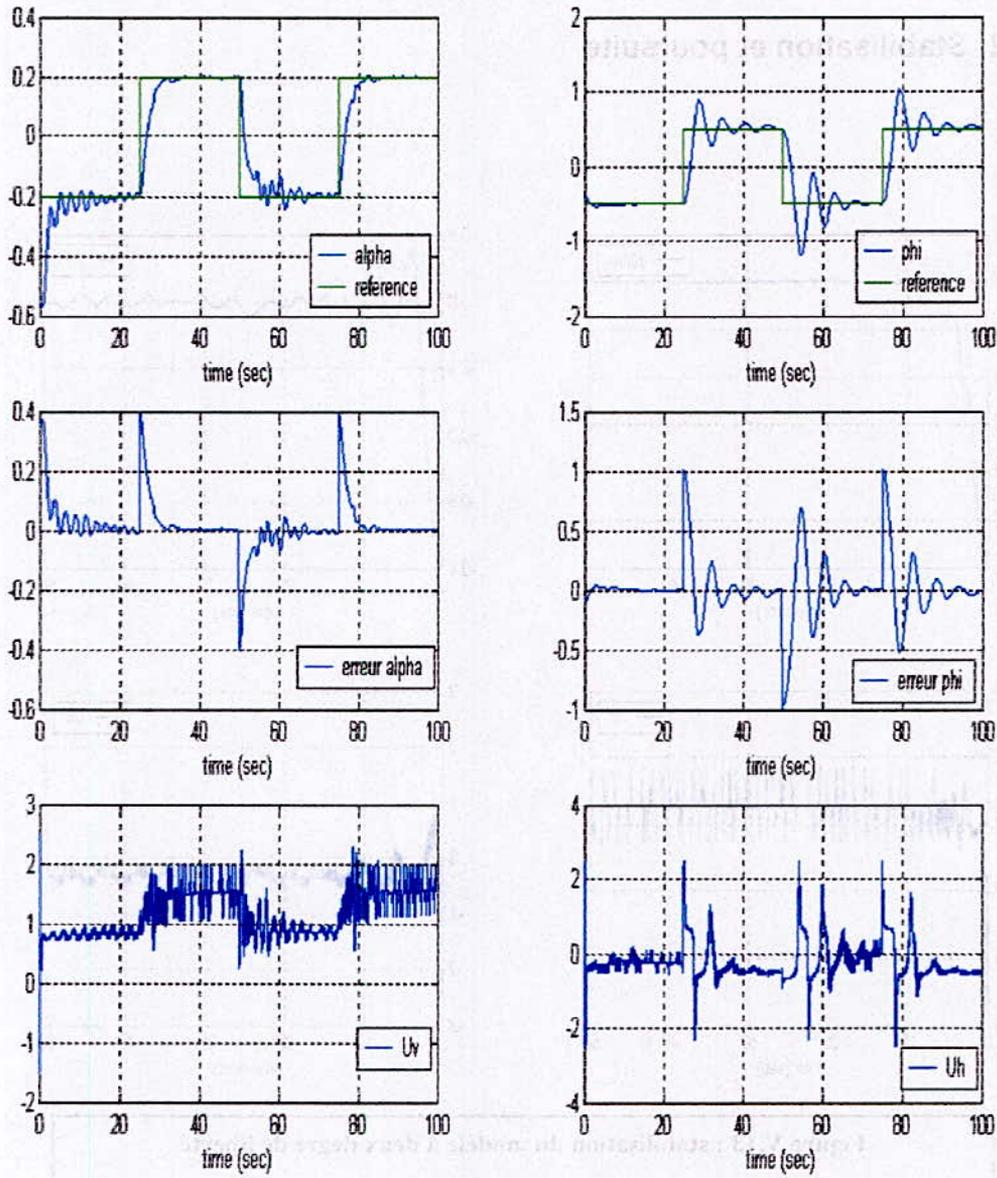


Figure V.14 : poursuite d'un signal carré

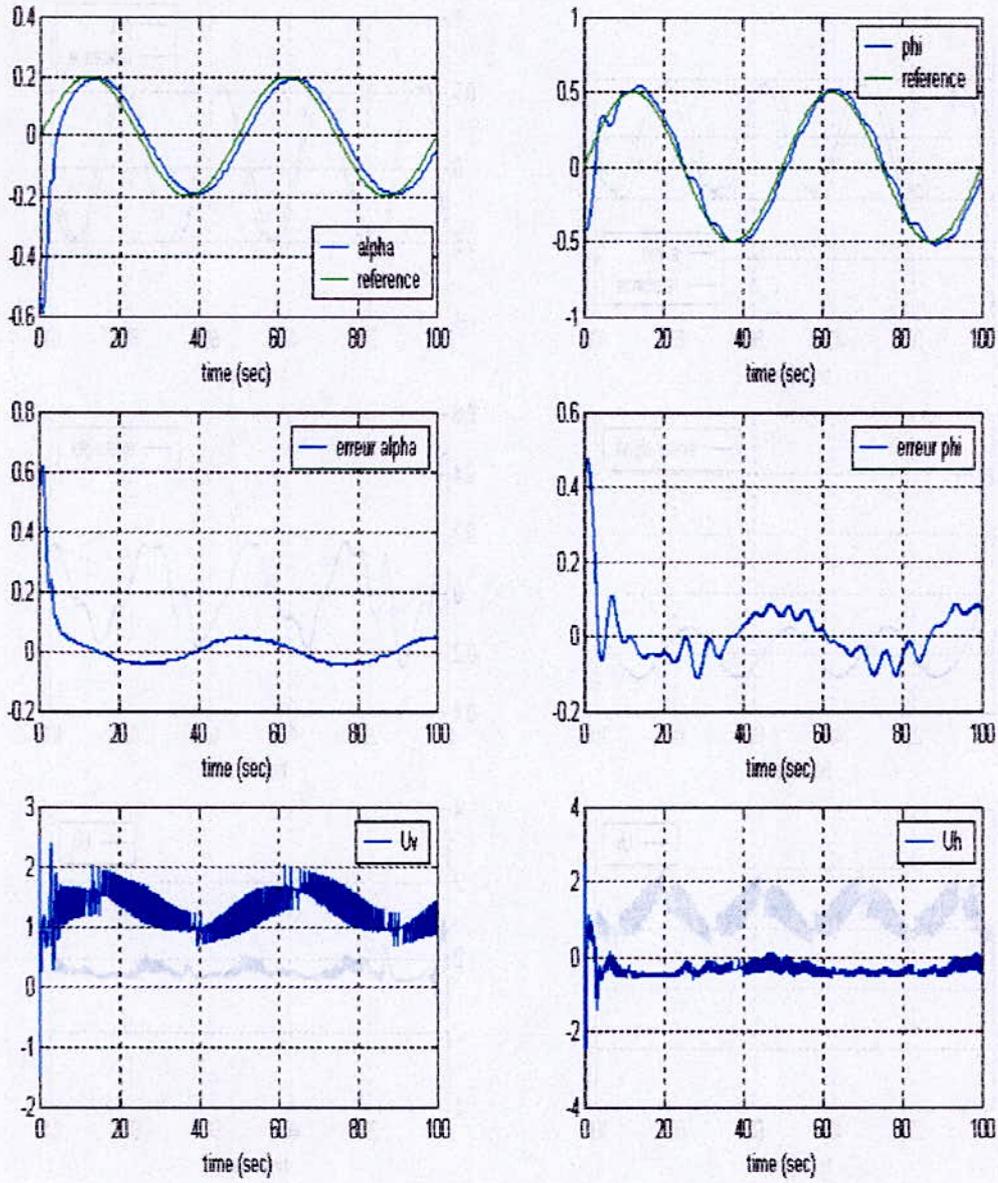


Figure V.15 : poursuite d'une sinusoïde (0.02 Hz)

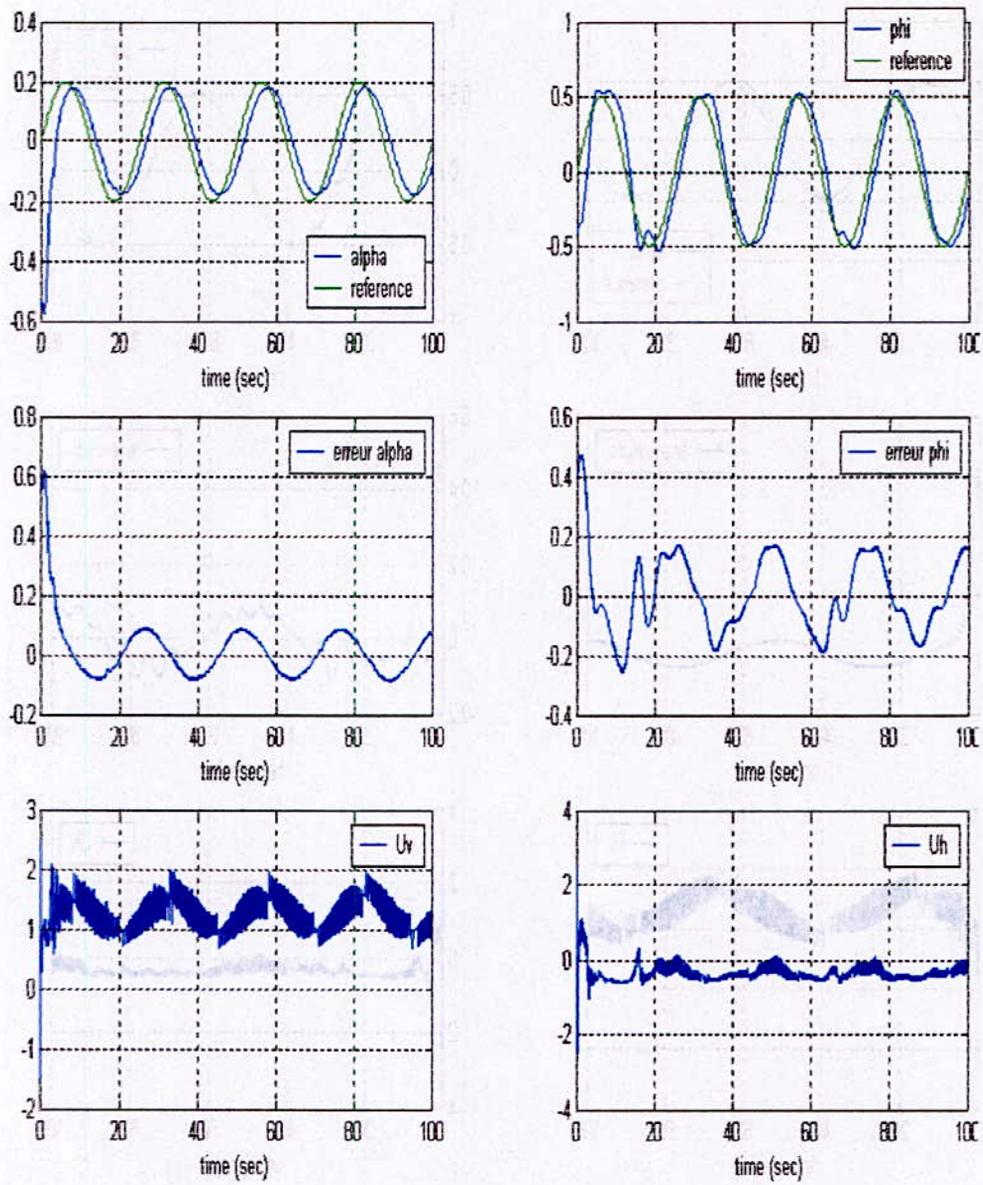


Figure V.16 : poursuite d'une sinusoïde (0.04Hz)

V.4.3 Variation paramétrique

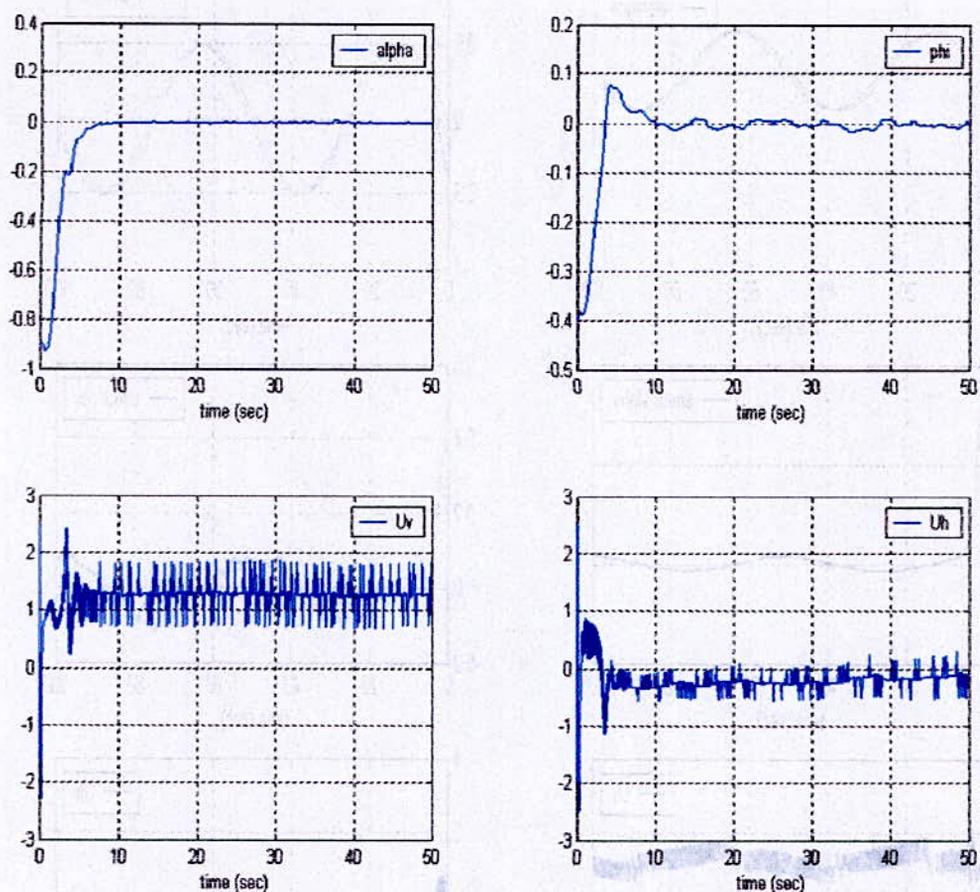


Figure V.17 : stabilisation avec introduction d'une variation paramétrique

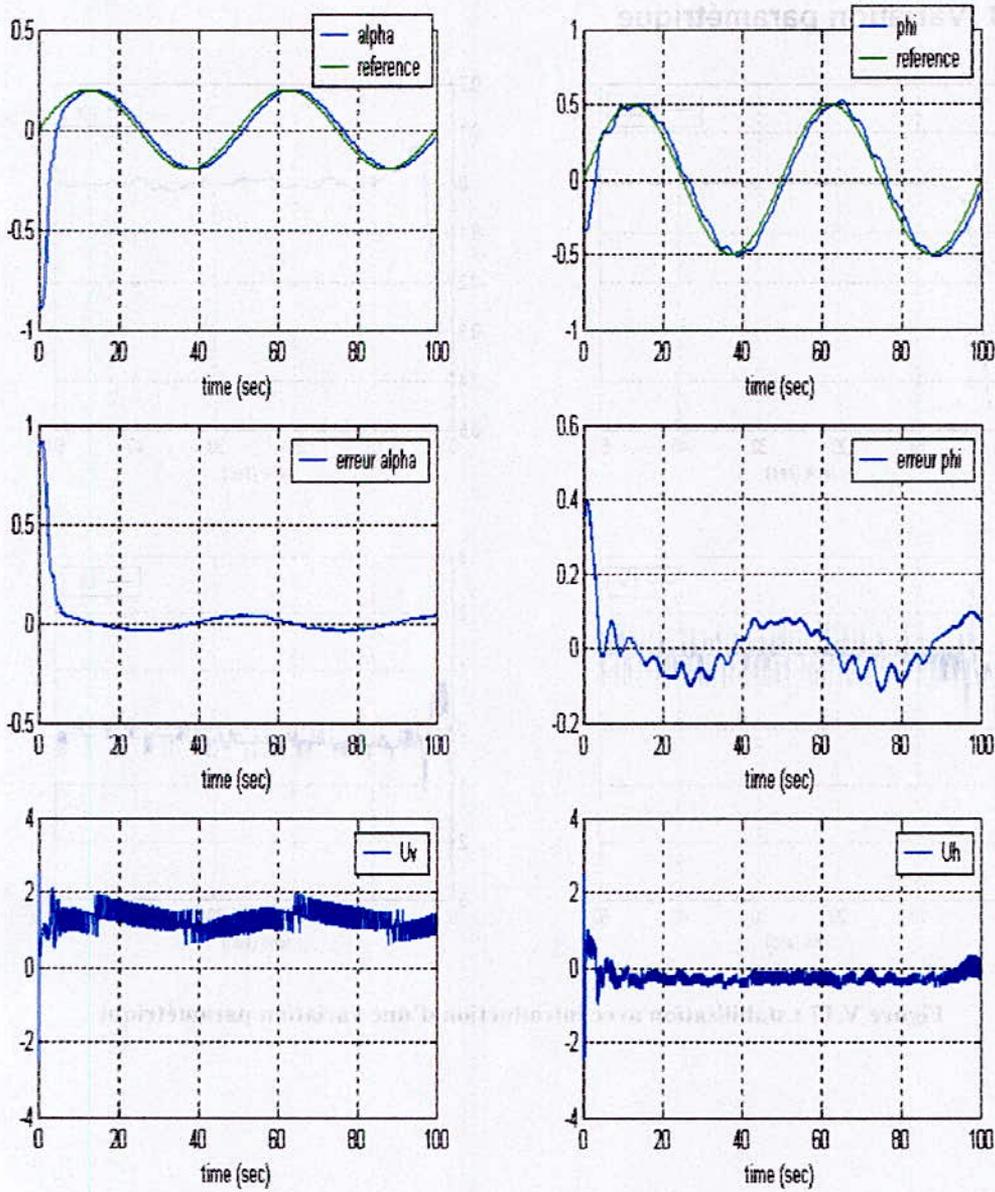


Figure V.18 : poursuite avec introduction variation paramétrique

V.4.4 Rejet de perturbation

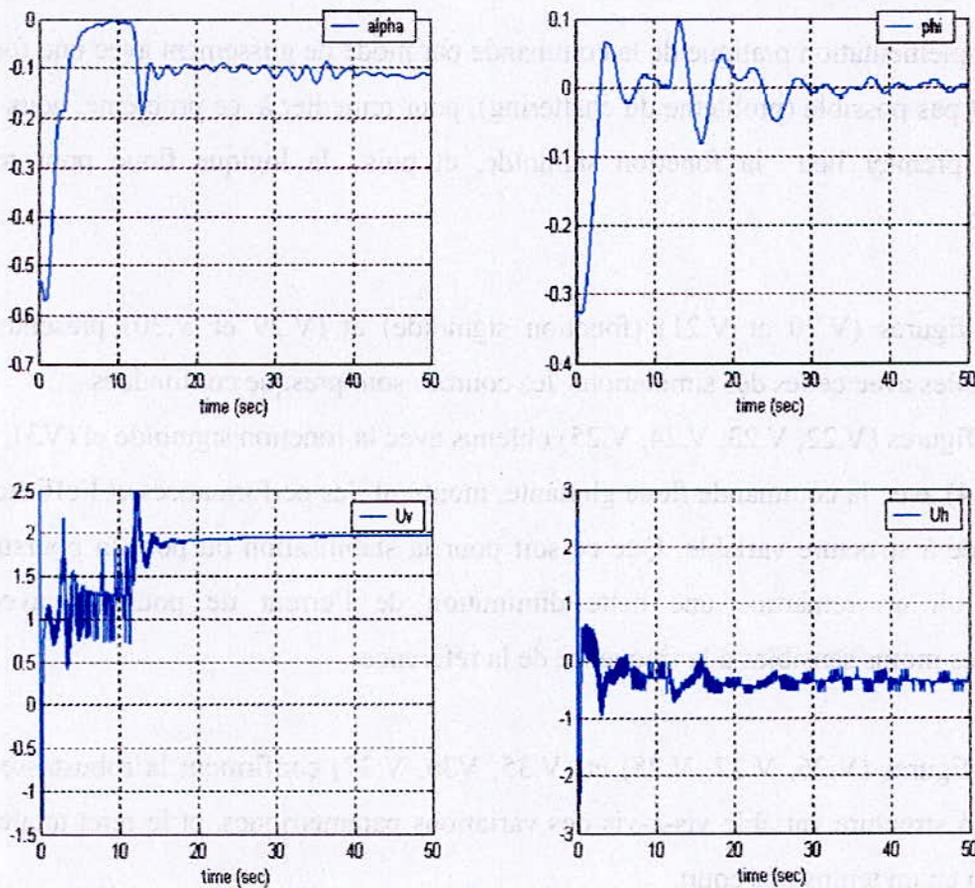


Figure V.19 : rejet de perturbation

Les limites de cette commande constatée lors du rejet de perturbation, nous ont amené à faire appel à une commande plus robuste dite « commande à structure variable ». Les résultats obtenus sont Présentées ci-après :

V.5 COMMANDE PAR MODE DE GLISSEMENT

L'implémentation pratique de la commande par mode de glissement avec une fonction signe, n'est pas possible (problème du chattering), pour remédier à ce problème, nous avons utilisé au premier lieu la fonction sigmoïde, et puis la logique floue pour réaliser l'attractivité.

Les figures (V.20 et V.21) (fonction sigmoïde) et (V.29 et V.30) présentent les réponses réelles avec celles des simulations, les courbes sont presque confondues.

Les figures (V.22, V.23, V.24, V.25) obtenus avec la fonction sigmoïde et (V.31, V.32, V.33 et V.34) avec la commande floue glissante, montrent les performances et l'efficacité de la commande à structure variable. Que ce soit pour la stabilisation ou pour la poursuite de trajectoire, où on remarque une nette diminution de l'erreur de poursuite avec des performances moins sensibles à la fréquence de la référence.

Les figures (V.26, V.27, V.28) et (V.35, V.36, V.37) confirment la robustesse de la commande à structure variable vis-à-vis des variations paramétriques, et le rejet totale de la perturbation en un temps très court

On a constaté que l'introduction de la logique floue, a apportée une légère amélioration des performances et de robustesse. Tout en sollicitant moins, les actionneurs.

V.5.1 Fonction d'attractivité : Sigmoidé

V.5.1.1 Réponse réelle VS réponse du modèle

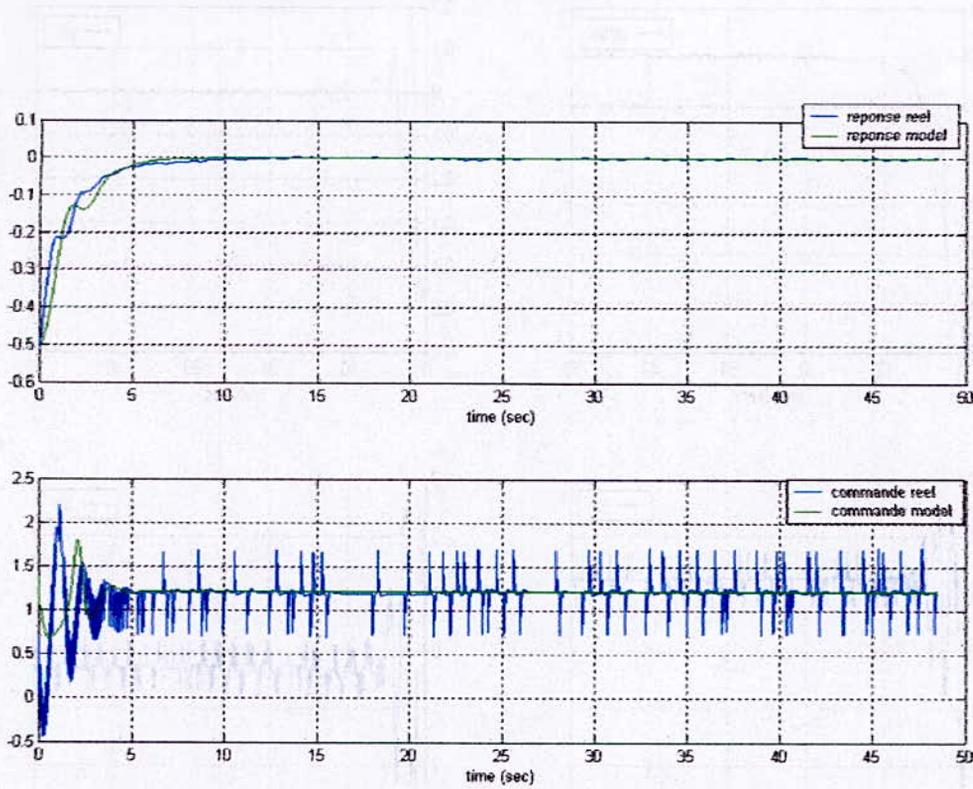


Figure V.20 : réponse modèle de simulation VS réponse réelle (modèle vertical)
Fonction d'attractivité : sigmoïde

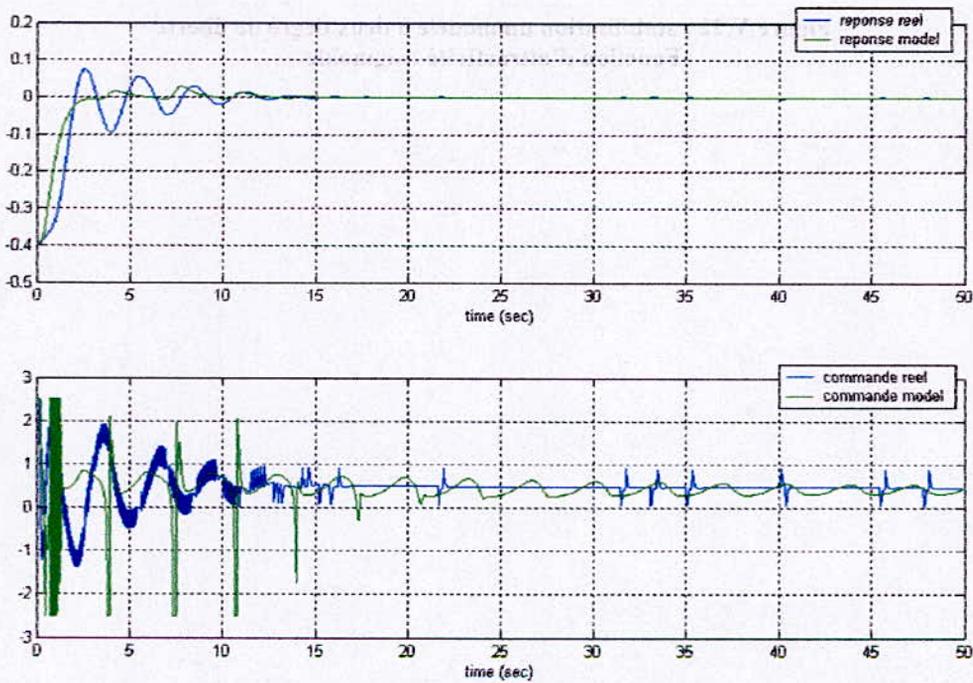


Figure V.21 : réponse modèle de simulation VS réponse réelle (modèle horizontal)
Fonction d'attractivité : sigmoïde

V.5.1.2 Stabilisation et poursuite

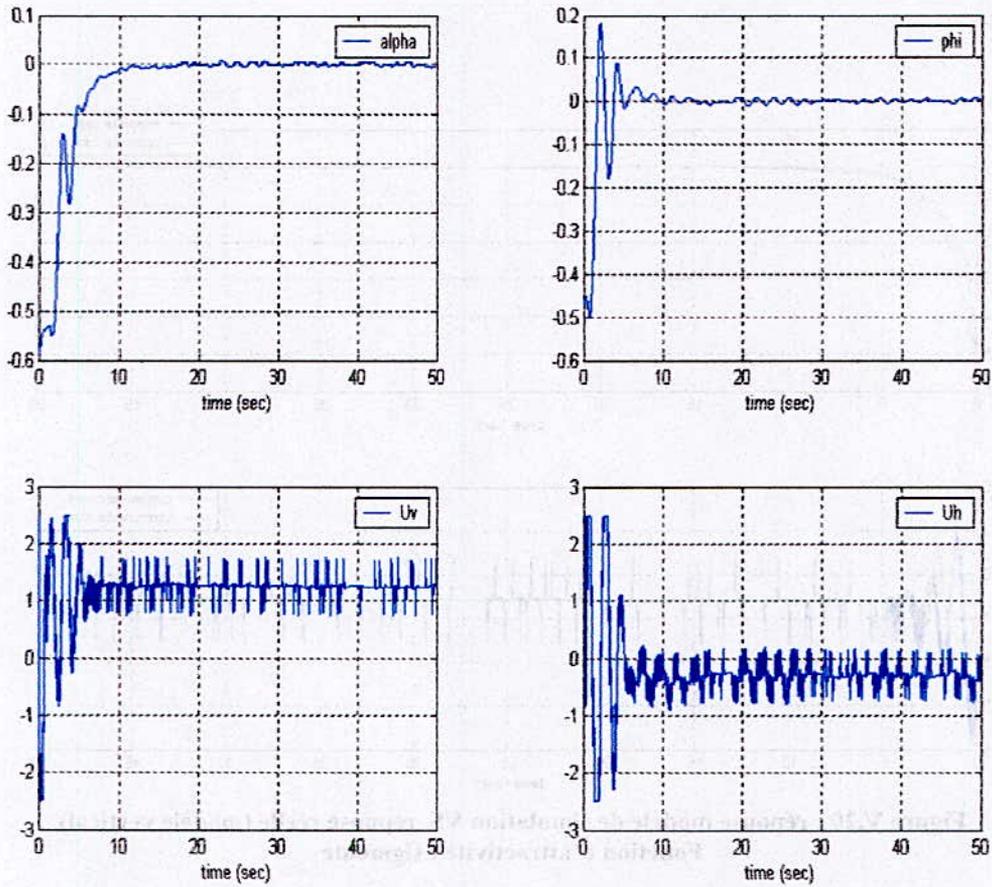


Figure V.22 : stabilisation du modèle à deux degré de liberté
 Fonction d'attractivité : sigmoïde

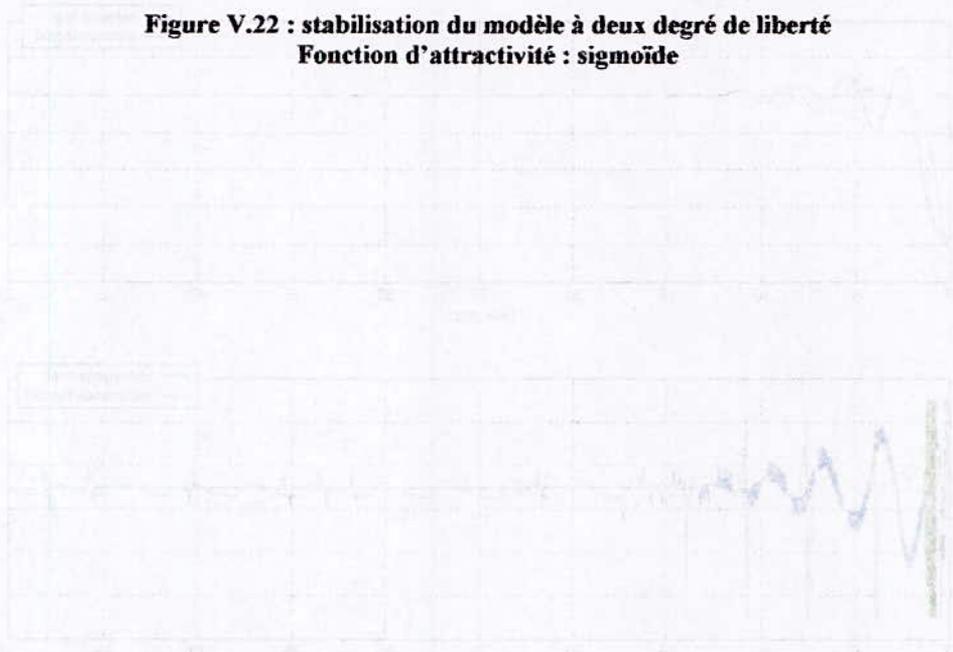


Figure V.23 : réponse modèle de simulation VS réponse réelle (modèle physique)
 Fonction d'attractivité : sigmoïde

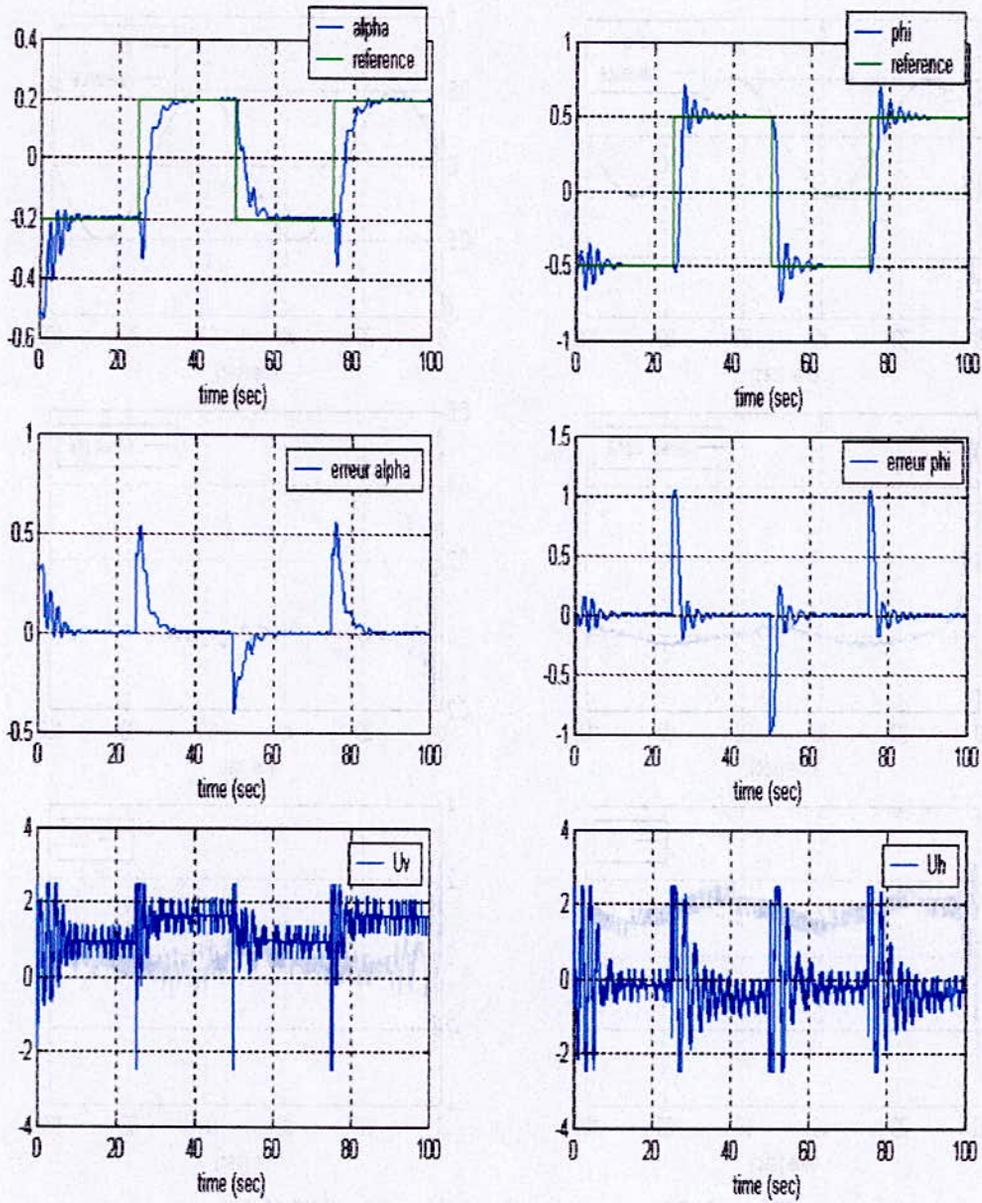


Figure V.23 : poursuite d'un signal carré
Fonction d'attractivité : sigmoïde

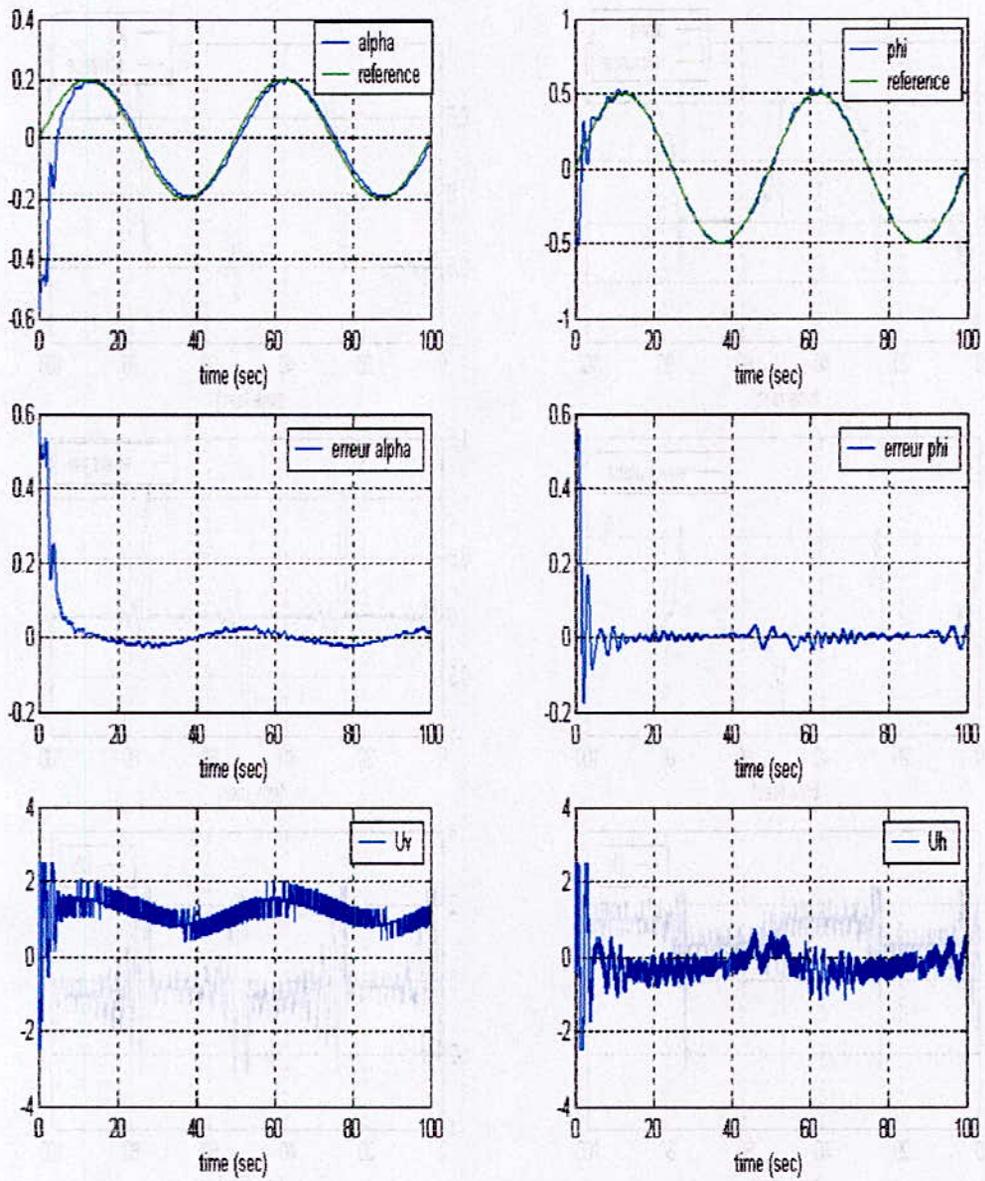
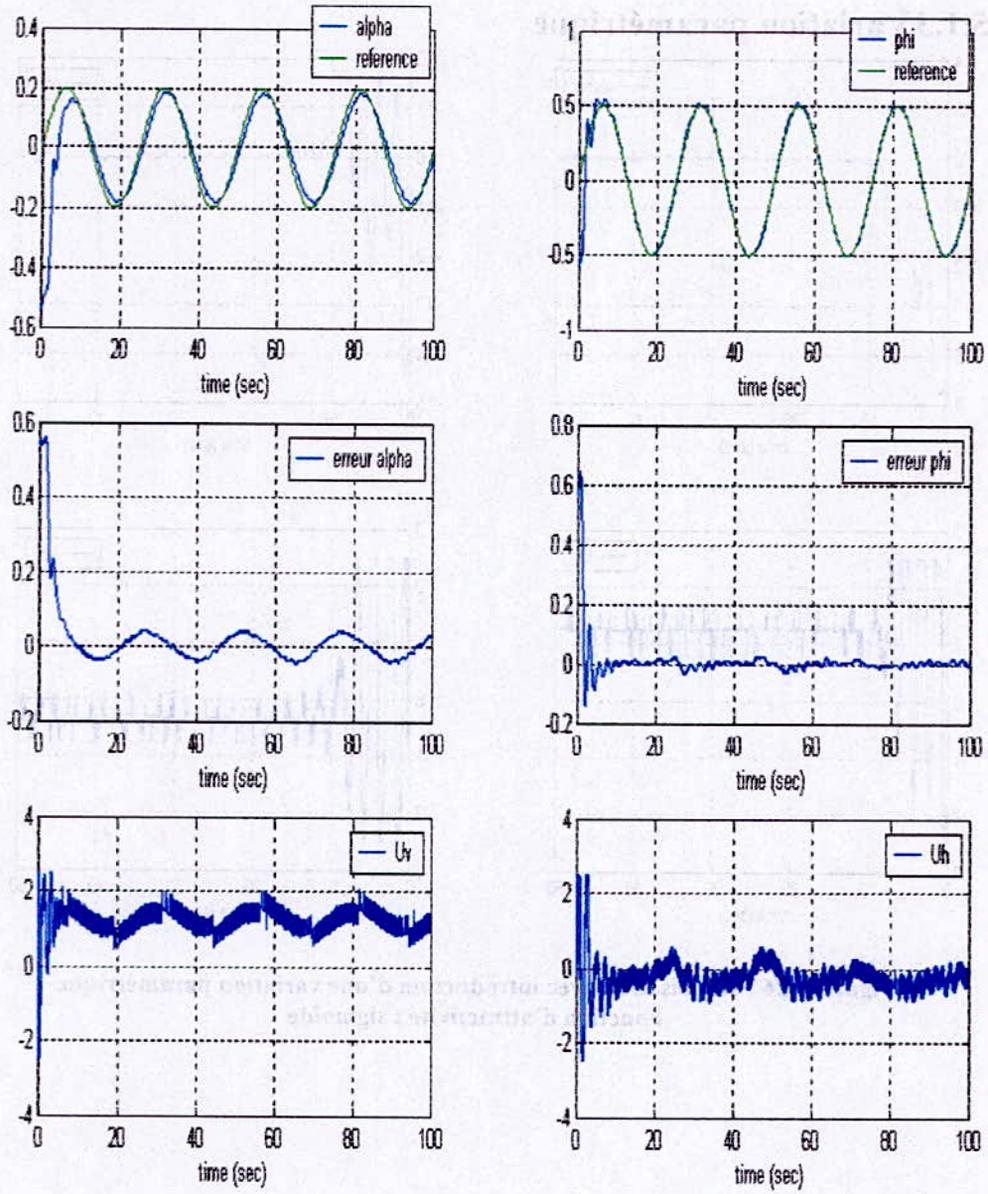


Figure V.24 : poursuite d'une sinusoïde (0.02 Hz)
Fonction d'attractivité : sigmoïde



**Figure V.25 : poursuite d'une sinusoïde (0.04Hz)
Fonction d'attractivité : sigmoïde**

V.5.1.3 Variation paramétrique

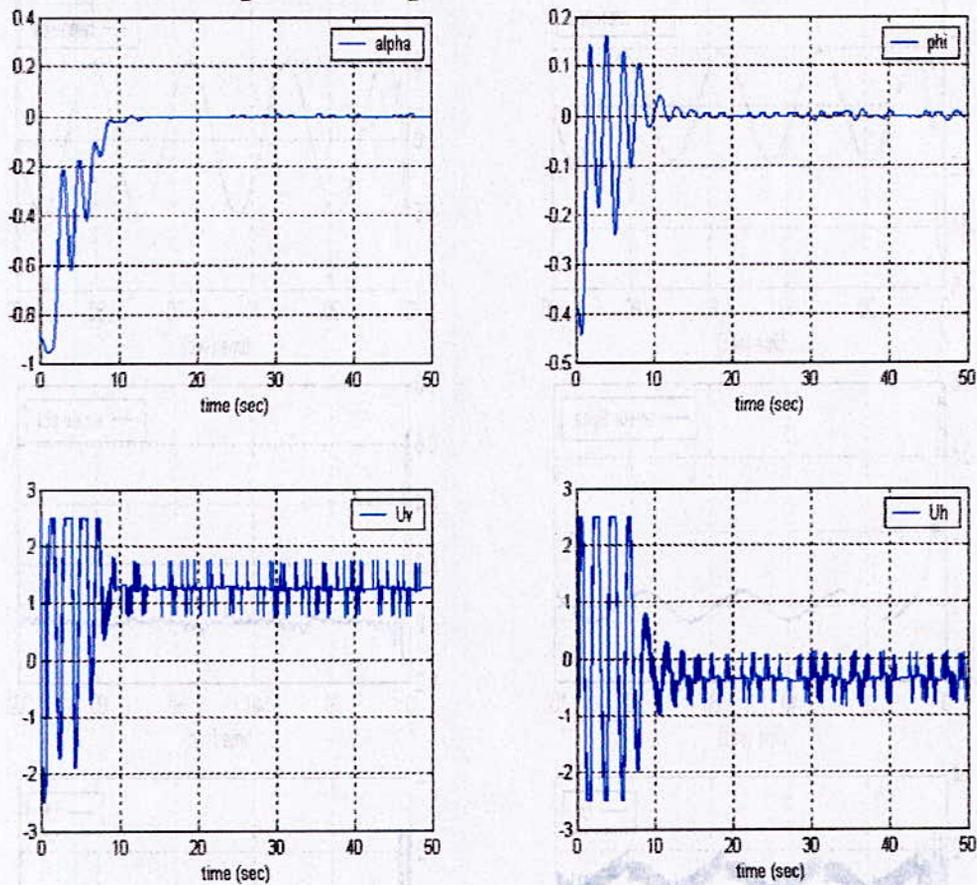
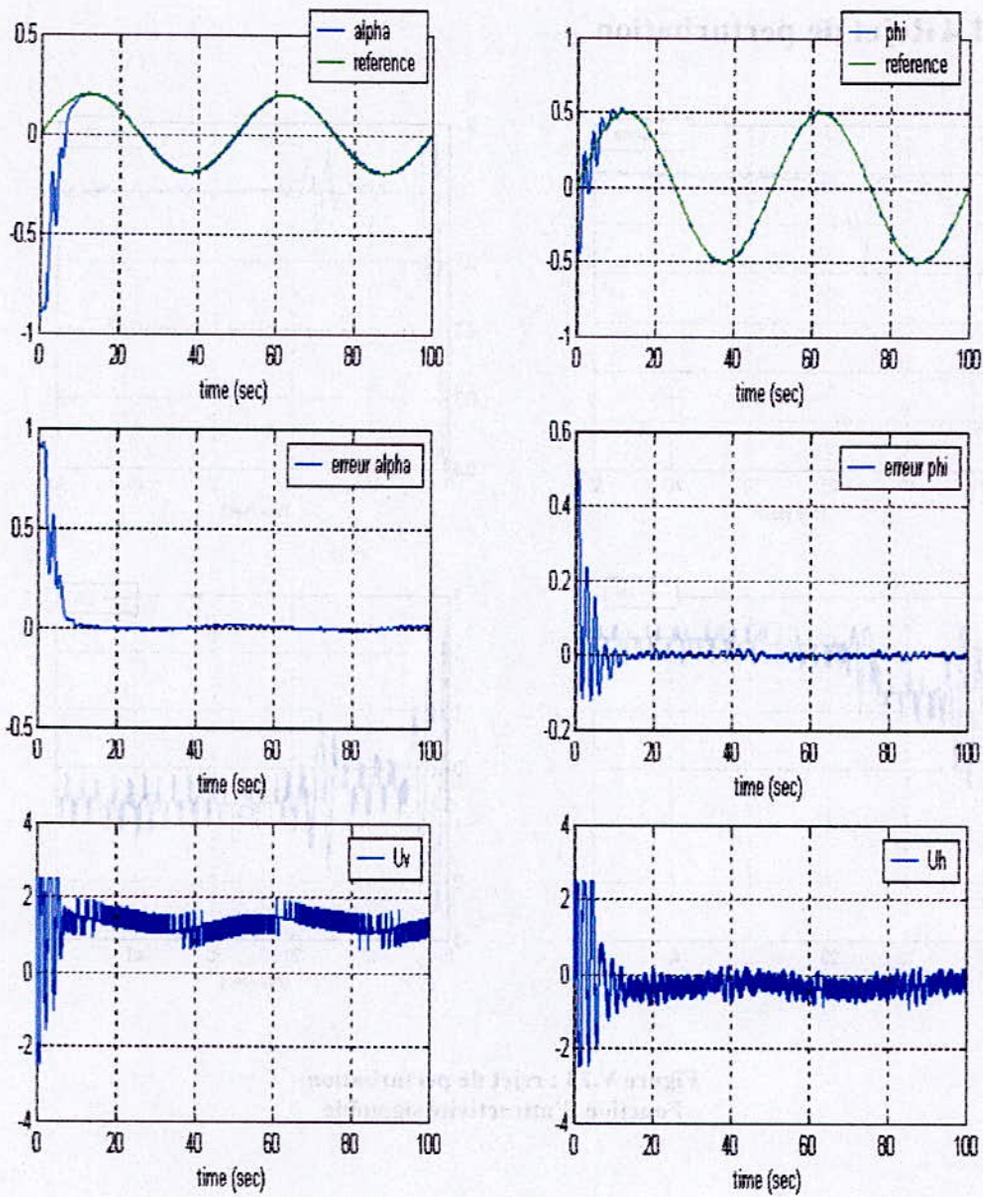


Figure V.26 : stabilisation avec introduction d'une variation paramétrique
Fonction d'attractivité : sigmoïde



**Figure V.27 : poursuite avec introduction d'une variation paramétrique
Fonction d'attractivité sigmoïde**

V.5.1.4 Rejet de perturbation

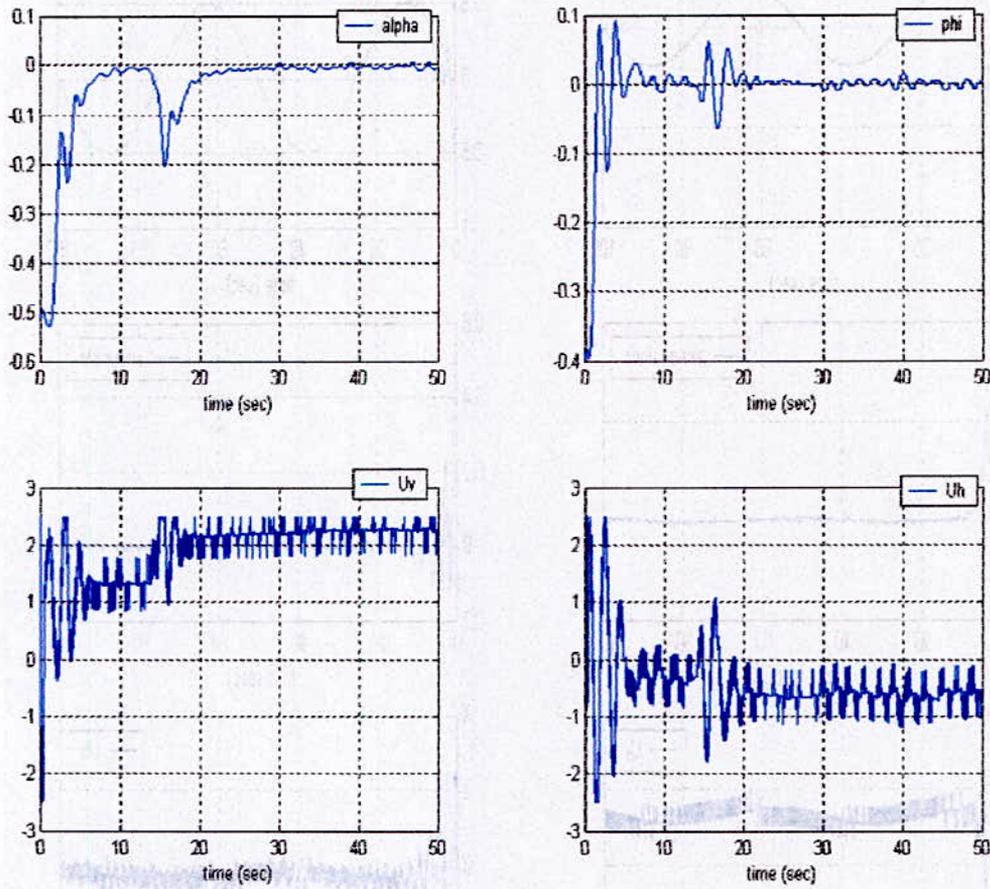


Figure V.28 : rejet de perturbation
Fonction d'attractivité sigmoïde

V.5.2 Flou glissant

V.5.2.1 Réponse réelle VS réponse du modèle

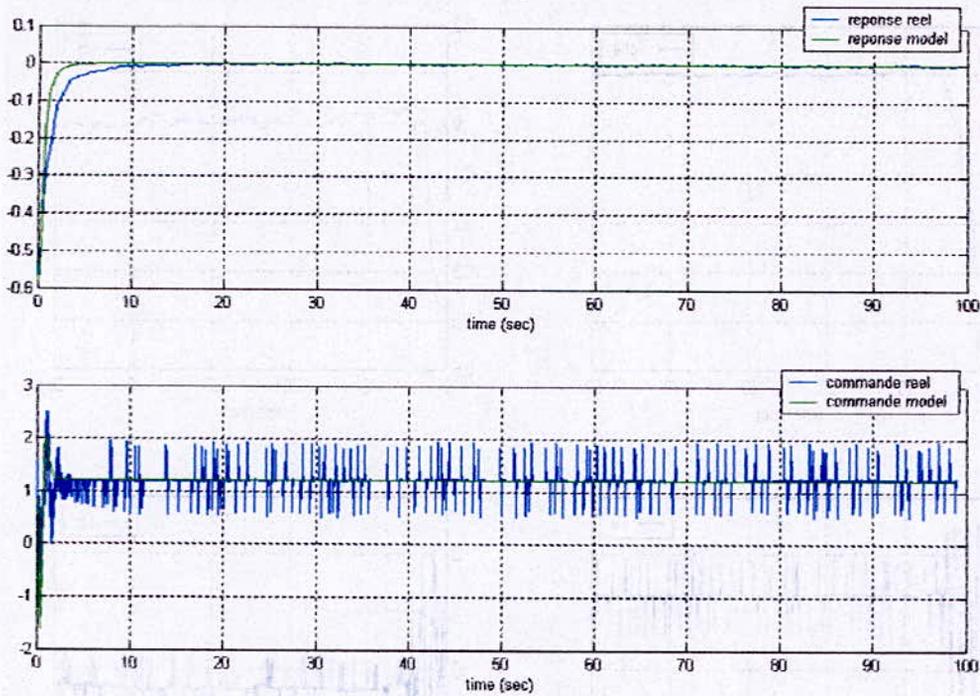


Figure V.29 : réponse du modèle de simulation VS réponse réelle (modèle vertical)
Fonction d'attractivité : bloc flou

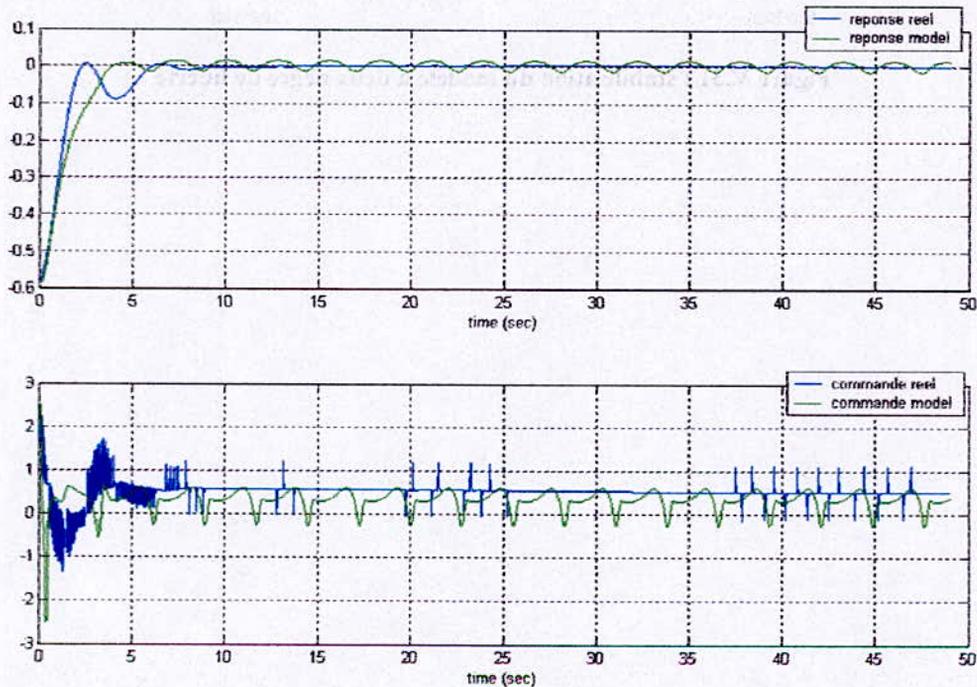


Figure V.30 : réponse modèle de simulation VS réponse réelle (modèle horizontal)
Fonction d'attractivité : bloc flou

V.5.2.2 Stabilisation et poursuite

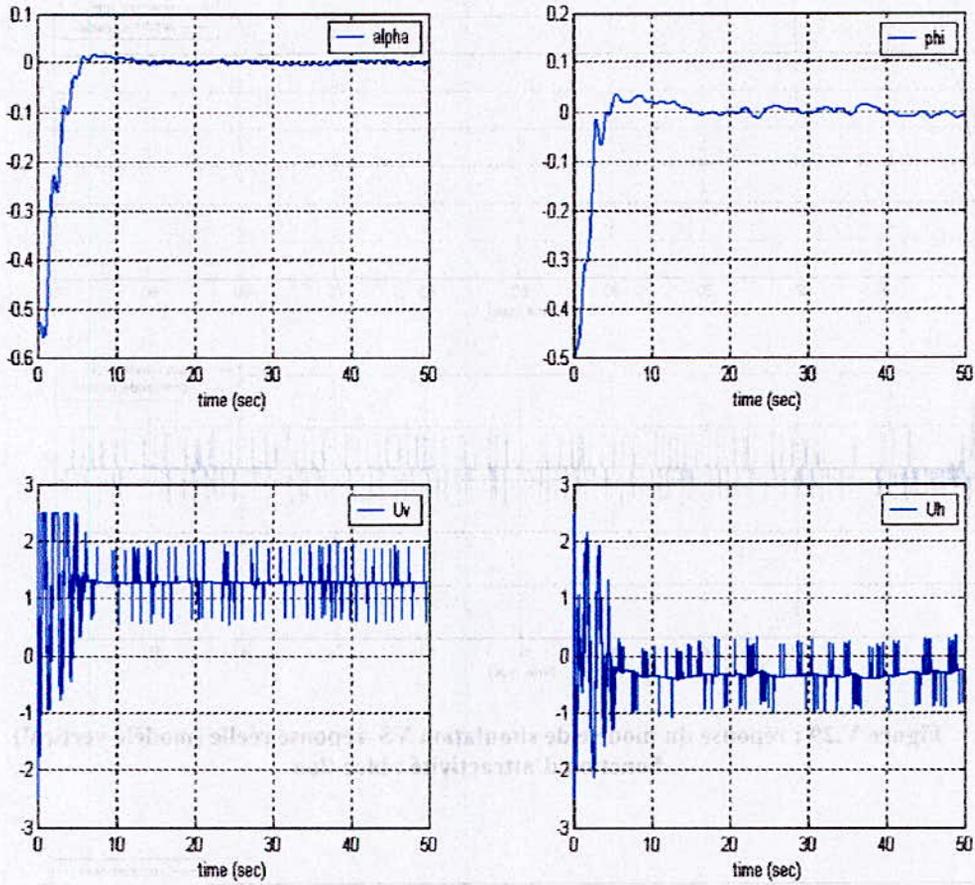


Figure V.31 : stabilisation du modèle à deux degré de liberté

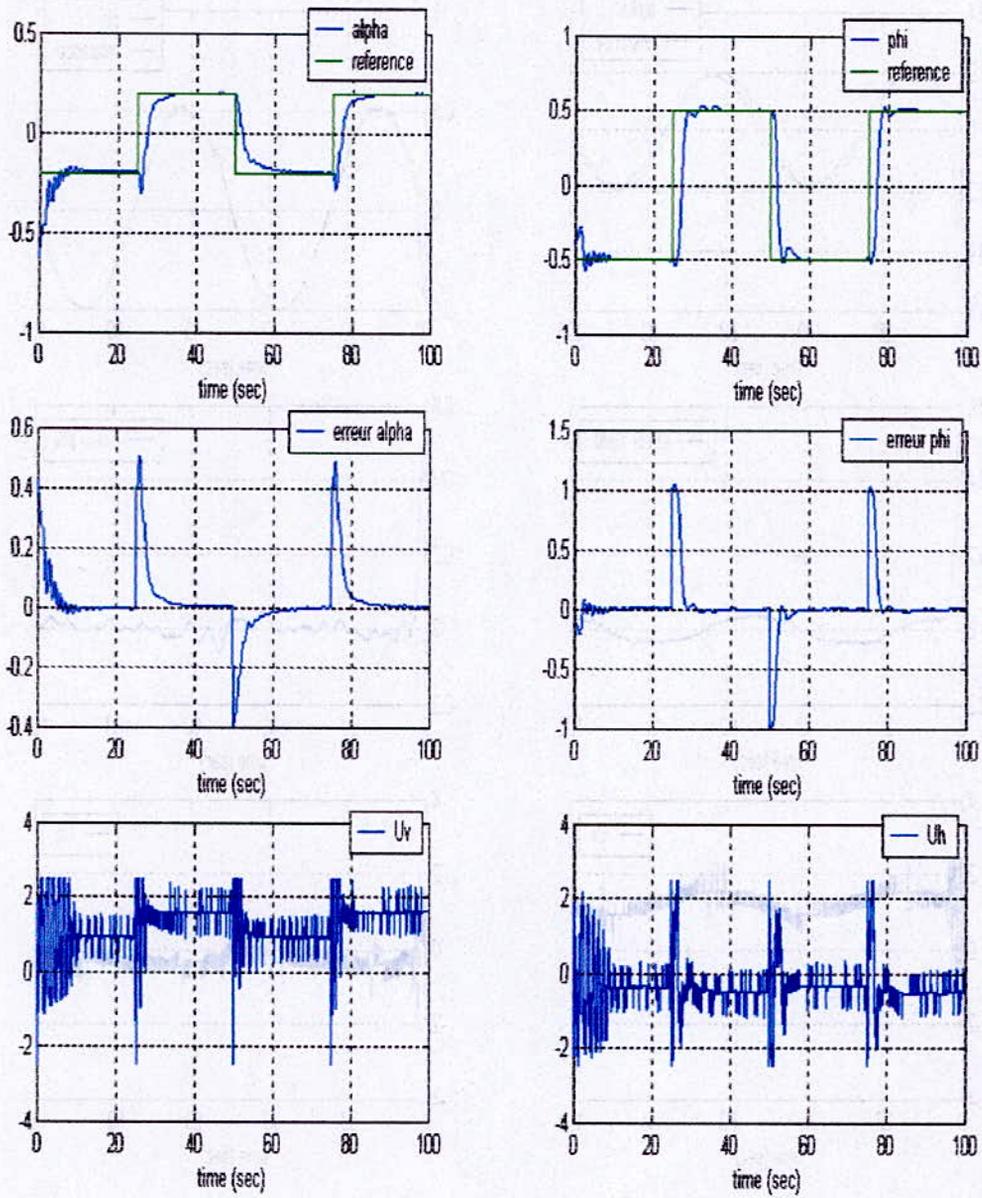


Figure V.32 : poursuite d'un signal carré

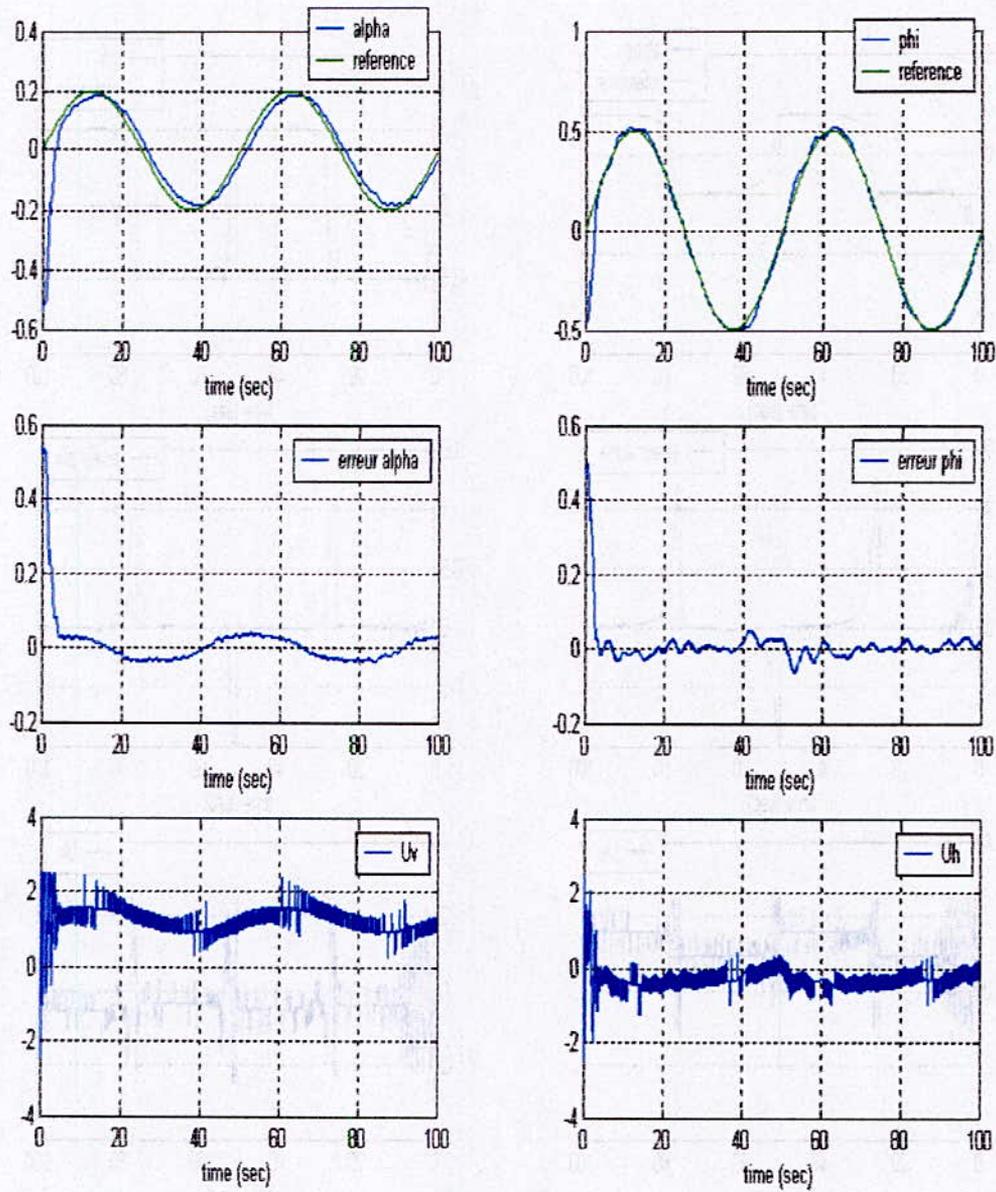


Figure V.33 : poursuite d'une sinusoïde (0.02 Hz)

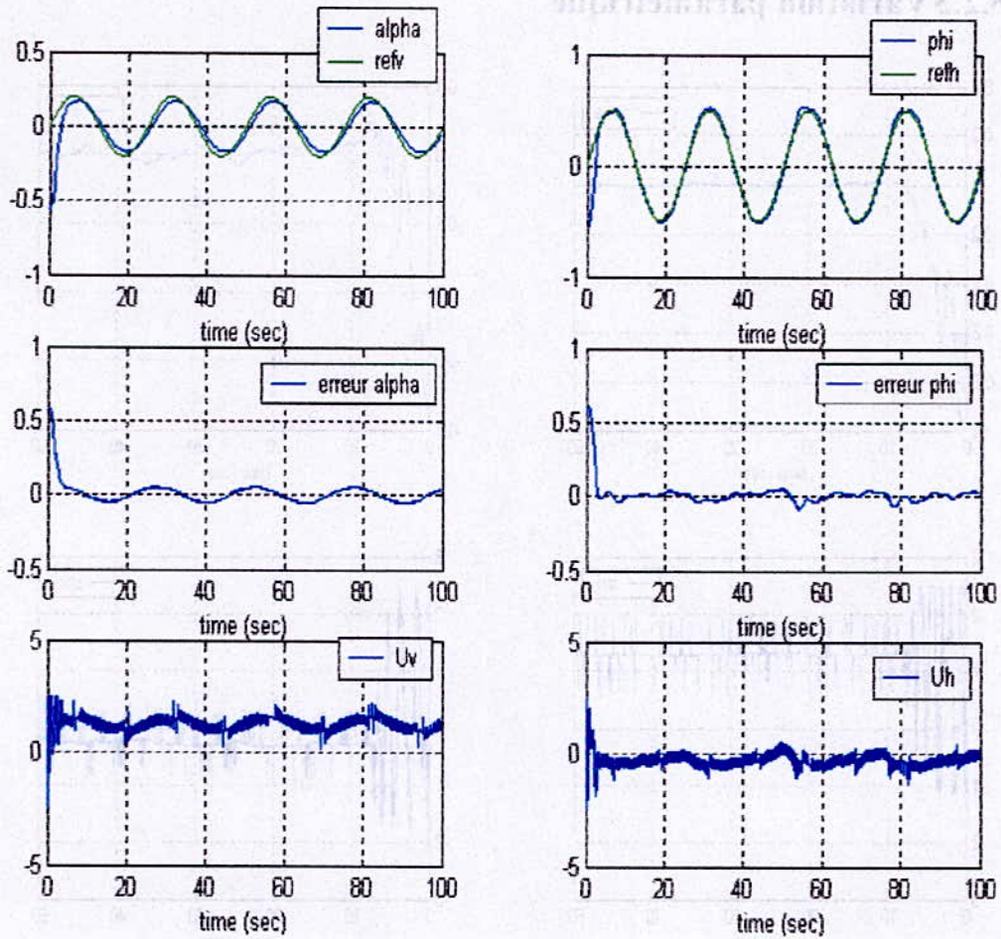


Figure V.34 : poursuite d'une sinusoïde (0.04 Hz)

V.5.2.3 Variation paramétrique

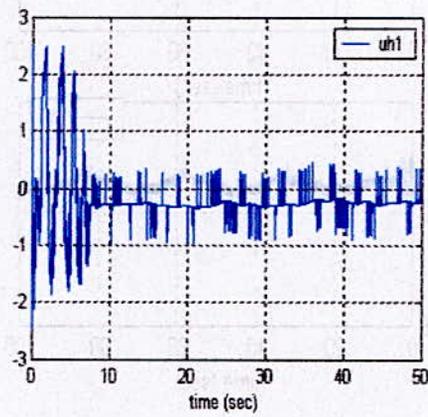
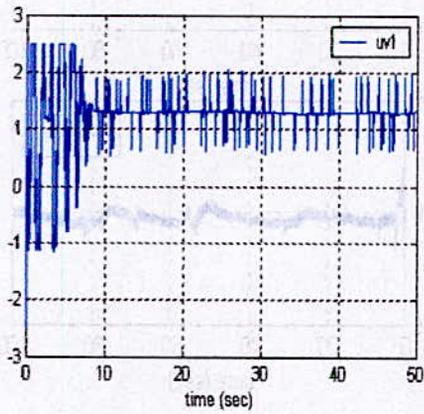
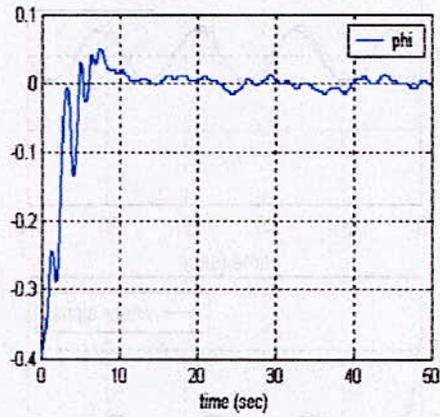
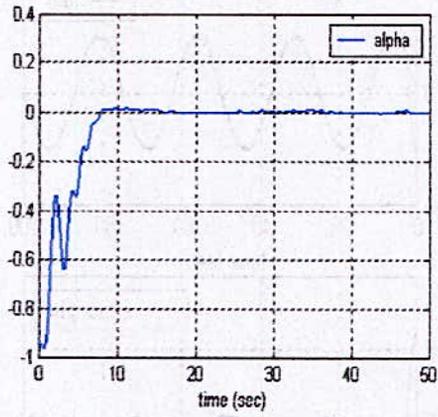


Figure V.35 : stabilisation avec variation paramétrique

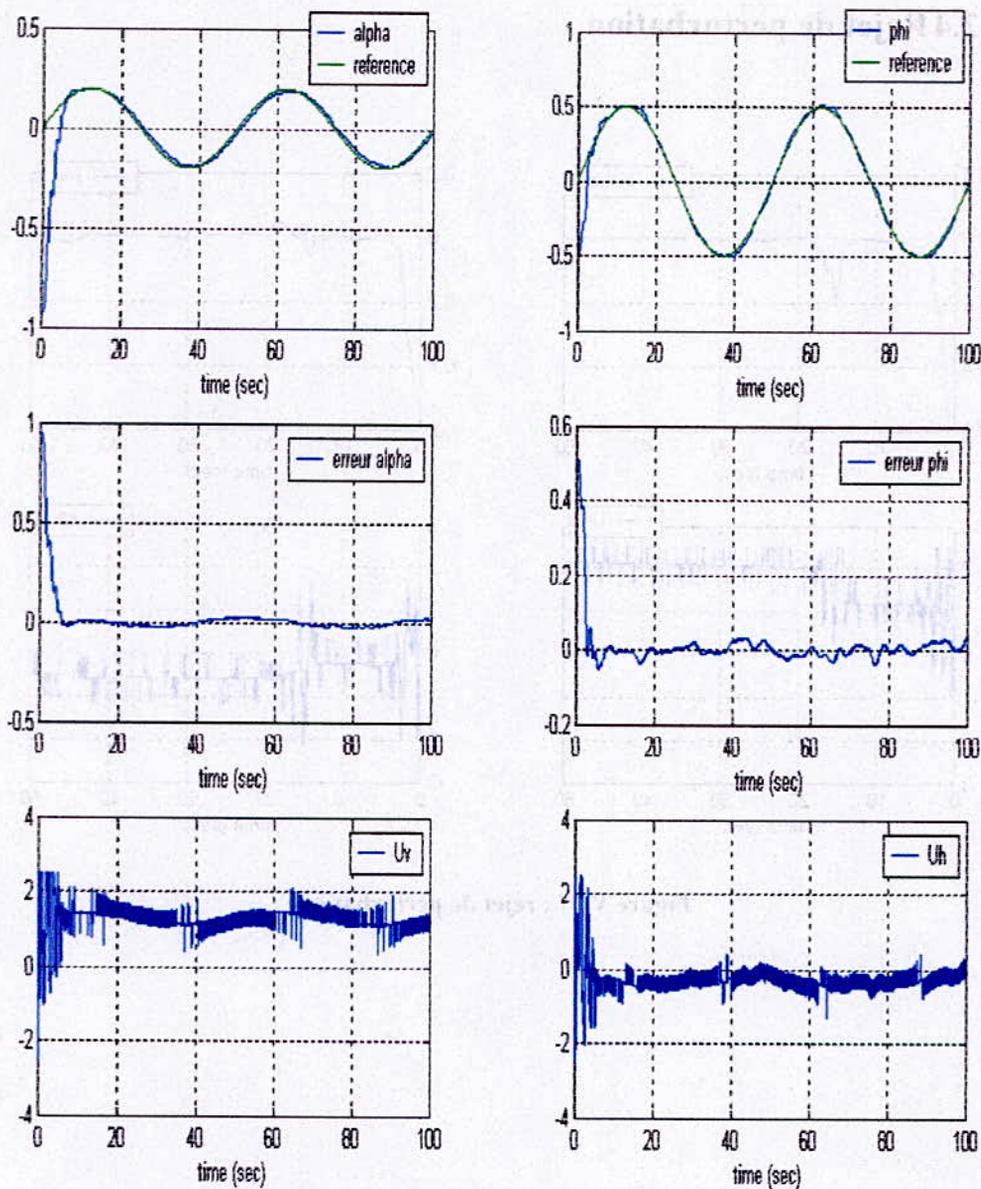


Figure V.36 : poursuite avec variation paramétrique

V.5.2.4 Rejet de perturbation

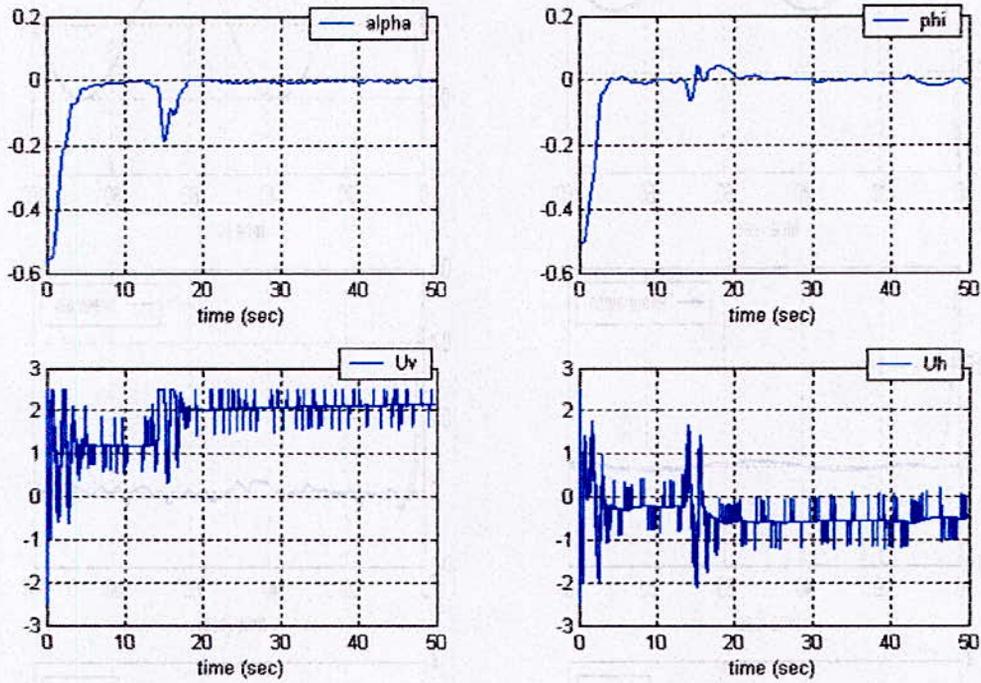


Figure V.37 : rejet de perturbation

V.6 ETUDE COMPARATIVE

A fin de faciliter la comparaison entre les différentes lois de commandes, nous avons quantifié les résultats obtenus précédemment dans des tables [tableau 1 et 2 et 3].

		Commande par mode de glissement	Commande floue glissante	Commande par bouclage linéarisant	Commande floue
Temps de réponse (sec)	Vertical	6.4850	4.32	5.0920	11.53
	Horizontal	4.585	3.8	11.1520	10.88
Maximum de l'erreur de poursuite (rad)	Vertical	0.0299	0.0404	0.0488	0.0717
	Horizontal	0.0348	0.0623	0.1056	0.0437
Variation de l'erreur de poursuite (rad)	Vertical	0.0162	0.0158	0.0395	0.0560
	Horizontal	0.0023	0.0101	0.0858	0.0981
Energie de commande (v^2)	Vertical	$1.8688 \cdot 10^4$	$1.8963 \cdot 10^4$	$1.4927 \cdot 10^4$	$1.7360 \cdot 10^4$
	Horizontal	$0.8145 \cdot 10^4$	$0.4968 \cdot 10^4$	$0.3265 \cdot 10^4$	$0.7545 \cdot 10^4$

table V.1 : Comparaison des performances

		Commande par mode de glissement	Commande floue glissante	Commande par bouclage linéarisant	Commande floue
Temps de réponse (sec)	Vertical	8.4660	6.8	5.3350	14.26
	Horizontal	8.6650	4.64	5.6490	13.24
Maximum de l'erreur de poursuite (rad)	Vertical	0.0190	0.0273	0.0418	0.0543
	Horizontal	0.0151	0.0386	0.1162	0.0620

Table V.2 : Variation paramétrique

		Commande par mode de glissement	Commande floue glissante	Commande par bouclage linéarisant	Commande floue
Effet de perturbation (rad)		0.2	0.19	0.1933	0.325
Temps de rejet de perturbation (sec)	Vertical	4	3	inf	6.5
Energie de commande (v^2)	Vertical	$2.0104 \cdot 10^4$	$1.8034 \cdot 10^4$	$1.6248 \cdot 10^4$	$1.6337 \cdot 10^4$
	Horizontal	$0.2823 \cdot 10^4$	$1.6917 \cdot 10^4$	$0.0827 \cdot 10^4$	$0.2567 \cdot 10^4$

Table V.3 : Rejet de perturbation

On constate que les commandes à structure variable (par mode de glissement et flou glissant) sont les plus performantes, et les plus robustes, d'autre coté ces commandes sont les plus énergétiques, de point de vue commande.

On remarque aussi que, la commande verticale est plus énergétique à celle de l'horizontal, à cause des forces gravitationnelles.

V.7 CONCLUSION

Dans ce chapitre, nous avons présenté les résultats expérimentaux des différentes lois de commandes. Nous avons vu que la commande par logique floue n'as pas donné de bons résultats, et que la commande linéarisante, est limitée en robustesse, nous avons vu aussi que l'introduction d'une structure variable, rend le régulateur robuste vis-à-vis des perturbations. Et que l'utilisation de la logique floue permet d'éliminer le phénomène du chattering et d'augmenter la robustesse tout en diminuant l'énergie de commande

Conclusion générale

Le travail que nous avons présenté dans ce mémoire consiste d'abord à faire une étude théorique sur la modélisation du simulateur d'hélicoptère, qui est un système physique conçu pour l'étude théorique et l'implémentation de nouveaux algorithmes de commandes. Nous avons vu qu'il était difficile d'avoir un modèle de connaissance précis.

Ensuite, nous avons abordé la théorie de la commande : floue, bouclage linéarisant, et la commande par mode de glissement, ainsi que leur application sur les modèles du simulateur d'hélicoptère (synthèse et simulation).

Comme étape finale de notre travail, nous avons implémenté ces lois de commandes en temps réel. Nous avons d'abord appris que la commande par logique floue, ne se base pas sur un modèle mathématique pour la synthèse, néanmoins, elle exige une bonne expertise pour trouver les bons paramètres du régulateur.

Nous avons vu aussi que la commande par bouclage non linéaire, est basée sur le modèle du système. Par conséquent, elle n'est pas robuste, et elle a un mauvais effet de régulation, surtout dans notre cas (modèles imprécis).

Enfin, nous avons constaté que La commande par mode de glissement est une commande robuste vis-à-vis perturbations extérieures. Et que L'utilisation de la logique floue pour réaliser l'attractivité, permet d'éliminer le phénomène du chattering et d'augmenter la robustesse de la commande par mode glissant. Cependant la stabilité de la partie floue reste fastidieuse a analysé..

BIBLIOGRAPHIE

- [mam75] E.H. Mamdani, « *An experiment in linguistic synthesis with a fuzzy logic controller* », int. J. Man Mach. Studies, Vol. 7, N°1, pp 1-73, 1975.
- [isid03] Alberto Isidori, 3rd edition, 2003, nonlinear control systems (communications and control engineering series)
- [sep01] Rodolphe Sepulchre, 2001-2002, Analyse des systèmes et introduction à leurs synthèses
- [slot92] J.J.E Slotine and Lee, Applied Nonlinear Control, 1992
- [man01] Feedback installation manual, 33-007-0
- [man02] feedback manual, 33-007-2
- [man04] feedback teaching manual 33-007-4
- [bald02] Jonas Balderud, Modeling and control of toy-helicopter, 2002
- [buh94] H. Bühler, « *Réglage par logique floue* », Presses polytechnique et université romandes, 1994.
- [jim93] M. Jimshidi, « *Fuzzy logic and control* », Ed Printice Hall, 1993
- [buhl86] H. Bühler, « *Réglage par mode de glissement* », Presses polytechnique et université romandes, 1986.
- [pau04] D.Arzelier-D.Paucelle « *Systèmes et asservissements non linéaires* », note de cours, 2004

- [dec88]** R.A.Decarlo et S.H.Zac et G.P.Matthews, « *Variable structure control of nonlinear multivariable system :Atutorial* », Proceedings of IEEE, Vol 76, N°3, pp 212-232, 1988.
- [lahd94]** Tarek Lahdhiri cart-pendulum balancing problem using fuzzy logic Control
- [utk77]** V.I.Utkin , « *Variable structure systems with sliding mode* », IEEE Transactions on Automatic control, Vol. Ac, N°2, April 1977.
- [lin94]** Sinn-Cheng Lin and Yung-Yaw Chen, design of adaptive fuzzy sliding mode for nonlinear system Control, IEEE 94
- [liao94]** Chung-Chun Kung and Chia-Chang Liao, Fuzzy-Sliding Mode Controller Design For Tracking Control Of Non-Linear System, Amadcan Control Confaranee, Balmom, Mayland June 1994
- [kafi99]** KAFI Mohamed Radouane, “Commande robuste et non linéaire: Application à un simulateur d’hélicoptère », 1999
- [rtw]** REAL TIME WORKSHOP, user’s manual
- [rtwintgt]** REAL TIME WINDOWS TARGET, user’s manual
- [bon02]** Christophe Bonivert, Synthèse d’un robot aeropropulsé, 2002
- [site]** www.helicoptere.net

Résumé :

L'objet de ce présent travail concerne l'implémentation pratique de lois de commandes non linéaires sur un simulateur d'hélicoptère : le TRMS (Twin Rotor Mimo System). Après présentation du banc d'essai, un modèle analytique non linéaire est développé. Ce dernier est adapté au modèle réel par une identification statique des paramètres. Par la suite, nous avons synthétisé quatre commandes, pour asservir l'angle d'élévation et d'azimut, qui sont : commande par logique floue, bouclage linéarisant, commande par mode de glissement et une commande hybride « flou glissant ». Enfin, ces commandes ont été appliquées en temps réel au TRMS, en faisant plusieurs expériences de tests et de validation.

Mots clés : simulateur d'hélicoptère, TRMS, Twin Rotor Mimo System, commande non linéaire, modes glissants, commande temps réel, logique floue, bouclage linéarisant

Abstract :

The aim of this project is the real time control of an bench-scale helicopter: The TRMS (Twin Rotor Mimo System). First, the bench-scale helicopter is presented, and then a considerable effort was required to develop a mathematical model that reflects the dynamics of the system, using static identification of parameters. Secondly, we have developed four control laws to control the pitch and yaw positions of the system, a fuzzy controller, feedback linearization controller, sliding mode controller and a fuzzy sliding mode controller. Then, these controllers were implemented in real time, several benches was made to validate them.

Key words: bench-scale helicopter, TRMS, Twin Rotor Mimo System, non linear control, sliding mode, fuzzy control, feedback linearization, real time control, fuzzy sliding mode

ملخص

يكمُن هدف هذا العمل في حساب و تطبيق مجموعة من قوانين التحكم اللاخطية على حاكي مروحية (نموذج مصغر لطائرة مروحية) ،بدءا بالتعريف بهذا النموذج ثم استخراج نموذج الرياضياتي و بعدها حساب قوانين التحكم و في الأخير تطبيقها على الحاكي.
الكلمات المفتاحية : حاكي مروحية ، التحكم اللاخطي ، المبدأ الإنزلاقي ، التحكم في زمن حقيقي ، المنطق الغامض ، التدوير اللاخطي.