



HAL
open science

Techniques floues et neuronales pour des problèmes de régulation

Dominique de Geest

► **To cite this version:**

Dominique de Geest. Techniques floues et neuronales pour des problèmes de régulation. Automatique / Robotique. INSA de Toulouse, 1995. Français. NNT : . tel-00144065

HAL Id: tel-00144065

<https://theses.hal.science/tel-00144065v1>

Submitted on 30 Apr 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Année 1995

THESE

Présentée
au Laboratoire d'Analyse et d'Architecture des Systèmes du CNRS

en vue de l'obtention du Titre
de DOCTEUR de l'Institut National des Sciences Appliquées de Toulouse

Spécialité **Automatique et Informatique Industrielle**

et du grade
de DOCTEUR EUROPEEN

par Dominique DE GEEST
Ingénieur INSAT

TECHNIQUES FLOUES ET NEURONALES POUR DES PROBLEMES DE REGULATION

Soutenue le 22 Septembre 1995 devant le jury:

M. L. Foulloy	Professeur (Université de Savoie)	Président
M. H. Bersini	Chargé de cours (Université libre de Bruxelles, Belgique)	Rapporteur
M. H.B. Verbruggen	Professeur-Ir (<i>Technische Universiteit Delft</i> , Pays Bas)	Rapporteur
M. S. Boverie	Chef de projet (Siemens Automotive, Toulouse)	Examineur
M. M. Flacher	Ingénieur (CORECI, Lyon)	Examineur
M. M. Jamshidi	Professeur (<i>University of New Mexico</i> , Etats Unis)	Examineur
M. A. Titli	Professeur (INSAT, Toulouse)	Directeur

Rapport LAAS N^o 95412

Thèse préparée au LAAS-CNRS

7, avenue du colonel Roche, 31077 Toulouse Cedex (France).

Avant-propos

Les travaux présentés dans ce mémoire ont été effectués à Toulouse au L.A.A.S. (Laboratoire d'Analyse et d'Architecture des Systèmes du C.N.R.S.). Par la présente préface, je tiens à remercier un certain nombre de personnes qui ont un lien avec mon travail.

Tout d'abord, je remercie *Mr André Titli* pour m'avoir recrutée comme doctorant, témoigné beaucoup de confiance tout au long des études menées avec beaucoup d'autonomie et offert des opportunités de nombreux contacts dans la communauté scientifique des techniques floues et neuronales pour l'Automatique. Je remercie également les directeurs de mes groupes d'accueil successifs *Mr Jacques Bernussou* et *Mme Louise Travé Massuyès* pour m'avoir permis de travailler dans leur équipe. Je remercie *Mr Alain Costes*, directeur du L.A.A.S., qui a accepté de m'accueillir dans son laboratoire.

Je suis reconnaissante aux membres du jury d'avoir accepté de juger mon travail de thèse. Je remercie donc les trois rapporteurs: *Mr Hugues Bersini*, Chargé de cours à l'Université libre de Bruxelles (Belgique), *Mr Laurent Foulloy*, Professeur à l'Université de Savoie et *Mr H.B. Verbruggen*, Professeur à l'Université technique de Delft (Pays Bas). Je remercie également les examinateurs de ma thèse: *Mr Serge Boverie*, Chef de projet à Siemens Automotives (Toulouse), *Mr Michel Flacher*, Ingénieur chez CORECI (Lyon), *Mr Mohamad Jamshidi*, Professeur à l'université du nouveau Mexique (Etats Unis).

Ensuite, je tiens à exprimer ma gratitude à toutes les personnes qui m'ont aidée ponctuellement ou avec qui j'ai collaboré pour des aspects techniques et généraux au cours de cette thèse. Je tiens, également, à remercier nos correspondants de la société CORECI (COMPAGNIE de REGULATION et de CONTRÔLE INDUSTRIEL, située à Lyon) *Mr Michel Flacher*, Ingénieur et *Mr Valancogne*, responsable de la division R&D, pour nous avoir fourni la possibilité de mener à bien une étude industrielle. Je remercie aussi les stagiaires français ou allemands avec qui j'ai travaillé et que j'ai encadrés: *Wolfgang Rehwald*, *Ernst Futterlieb* et *Hugues Jocteur Monrosier*.

Finalement, je réserve une pensée spéciale aux personnes, d'origine parfois lointaine, que j'ai pu côtoyer amicalement pendant ces quatre dernières années....

Merci à *Vincent* pour sa présence de tous les jours....

Préface

La vérité objective:

"Les concepts physiques sont des créations libres de l'esprit humain et ne sont pas, comme on pourrait le croire, uniquement déterminés par le monde extérieur. Dans l'effort que nous faisons pour comprendre le monde, nous ressemblons quelque peu à l'homme qui essaie de comprendre le mécanisme d'une montre fermée. Il voit le cadran et les aiguilles en mouvement, il entend le tic-tac, mais il n'a aucun moyen d'ouvrir le boîtier. S'il est ingénieux il pourra se former quelque image du mécanisme, qu'il rendra responsable de tout ce qu'il observe, mais il ne sera jamais sûr que son image soit la seule capable d'expliquer ses observations. Il ne sera jamais en état de comparer son image avec le mécanisme réel, et il ne peut même pas se représenter la possibilité ou la signification d'une telle comparaison. Mais le chercheur croit certainement qu'à mesure que ses connaissances s'accroissent, son image de la réalité deviendra de plus en plus simple et expliquera des domaines de plus en plus étendus de ses impressions sensibles. Il pourra aussi croire à l'existence d'une limite idéale de la connaissance que l'esprit humain peut atteindre. Il pourra appeler cette limite idéale la *vérité objective*."

Albert Einstein, Léopold Infeld, *L'évolution des idées en physique*
(Payot, 1978, p. 34-35).

"The no best model is fundamental to Engineering, whereas in Science, where task of modelling is almost exclusively analytic - to describe the physical world as accurately as possible - the notion of best model may be valid. In Engineering, concerned with synthesis as well as analysis:

a model is correct if it satisfies its purpose."

Roy Leitch, *Knowledge based control: selecting the right tool for the job*
(dans: Preprints of the IFAC/IFIP/IMACS International Symposium on Artificial Intelligence in Real-Time Control, Delft, Pays Bas, 16-18 Juin 1992, p. 25-33).

Table des matières

Liste des abréviations utilisées	4
Introduction	5
1 Position du problème	8
1.1 Intégration de systèmes de contrôle-commande	9
1.1.1 L'automatisation avancée	9
1.1.2 Limitations des approches conventionnelles	12
1.2 Les techniques connexionistes et floues	13
1.2.1 Les réseaux de neurones	13
1.2.2 Les techniques d'IA, et plus particulièrement le flou	15
1.3 Le bon outil pour un problème d'automatisation	18
1.3.1 Approche méthodologique	18
1.3.2 Des combinaisons de techniques	19
1.4 Objectifs du travail de thèse	19
1.4.1 Le pendule inversé	20
1.4.2 Les fours	23
2 Régulation floue: Méthodologies	25
2.1 Introduction aux techniques floues	26
2.1.1 Domaines scientifiques autour de la régulation floue	26
2.1.2 Présentation générale des régulateurs flous	34
2.2 Conception des régulateurs flous	37
2.2.1 Paramètres des convertisseurs N-F et F-N	37
2.2.2 Paramètres de la base de règles	39
2.2.3 Paramètres de la base de définitions des ensembles flous	41
2.2.4 Paramètres du moteur d'inférence	46
2.2.5 Bilan: ordre de réglage des paramètres du RF	47
2.3 Aspects théoriques et pratiques de mise en oeuvre	48
2.3.1 Les problèmes de stabilité et robustesse	48
2.3.2 Solutions proposées pour les problèmes d'imprécision	50

3	Régulation floue: Applications	55
3.1	Applications sur le pendule inversé	56
3.1.1	Modèle linéaire, comparaison avec la littérature	56
3.1.2	Modèle non linéaire, analyse de la loi de régulation	58
3.1.3	Modèle non linéaire, départ de la position basse	63
3.1.4	Applications sur le système réel	65
3.2	Applications sur les fours	69
3.2.1	Réglage des RF utilisés	69
3.2.2	Analyse des résultats	71
3.3	Conclusions et extensions	74
4	Régulation neuronale	76
4.1	Les réseaux de neurones: introduction	77
4.1.1	Dates dans l'étude des réseaux neuronaux artificiels	77
4.1.2	Les réseaux utilisant la rétropropagation	80
4.2	Conception du régulateur neuronal proposé	83
4.2.1	Description du régulateur neuronal	83
4.2.2	Loi de régulation neuronale proposée: régime glissant	88
4.2.3	Propositions pour des calculs optimaux des directions du gradient	90
4.2.4	Mise en oeuvre: choix des paramètres du réseau	96
4.3	Application au pendule inversé	97
4.3.1	Simulations	97
4.3.2	Système réel	100
4.4	Conclusions et extensions	102
5	Réglage d'un PID par logique floue	104
5.1	Définition des superviseurs flous	105
5.1.1	Principe général	105
5.1.2	Les deux approches principales	106
5.2	Mises en oeuvre des superviseurs flous	107
5.2.1	Régulateurs PID et superviseur considérés	107
5.2.2	Stratégies d'adaptation utilisées pour les gains du PID	110
5.2.3	Choix effectués pour les paramètres du superviseur et Indications sur les liens avec les performances	113
5.3	Applications sur les fours	116
5.3.1	Application de la première stratégie	116
5.3.2	Application de la deuxième stratégie en simulation	120
5.4	Conclusions et extensions	122
6	Auto-réglage de régulateurs flous	124
6.1	Génération automatique de lois de régulation floue non linéaire	125
6.1.1	Justification d'une telle démarche	125

6.1.2	Sur quels paramètres du RF agir et comment ?	126
6.2	Mise en oeuvre d'un auto-régulateur de RF: démarche générale	127
6.2.1	Principe général de l'auto-réglage	127
6.2.2	Réglage d'un PI flou non linéaire	128
6.2.3	Choix des paramètres par l'utilisateur	129
6.3	Extension à une régulation adaptative: amélioration du module d'auto-réglage de base	131
6.3.1	Principe général	131
6.3.2	La table de règles	131
6.3.3	Les choix des paramètres	132
6.4	Application à la régulation de température de fours	132
6.4.1	Simulations d'auto-réglage de base	133
6.4.2	Simulations avec un auto-réglage autonome	141
6.4.3	Auto-réglage de base avec la réglette à infra rouge	143
6.5	Conclusions et extensions	145
	Conclusions et Perspectives	147
	Références	152
	Annexes	156
	A1- Equivalence exacte entre des lois de régulation floue et PI analytique .	156
	A2- Méthode de stabilité pour la supervision floue de régulateur PID . . .	159

Liste des abréviations utilisées

- CI:** Condition Initiale;
- EF:** Ensemble Flou, ou, plus précisément sous-ensemble flou;
- E-S:** Entrées-Sorties;
- F-N:** Flou-Numérique;
- IA:** Intelligence Artificielle;
- MP:** méthode de défuzzification par Moyenne Pondérée;
- N-F:** Numérique-Flou;
- nq:** niveaux de quantification;
- PD:** Proportionnel-Dérivé;
- PI:** Proportionnel-Intégral;
- PID:** Proportionnel-Intégral-Dérivé;
- RE:** régulateur à Retour d'État;
- RC:** Régulateur en Cascade;
- RF:** Régulateur Flou;
- RN:** Régulateur Neuronal;
- RPP:** Régulateur à Placement de Pôles;
- SF:** Superviseur Flou;
- SNCC:** Système Numérique de Contrôle Commande;
- $t_{re\%}$: Temps de réponse à $\epsilon\%$ (ϵ très petit).
-

Introduction

Ces dernières années s'est développé un intérêt particulier pour les techniques floues et neuronales. Les systèmes experts flous permettent principalement de traiter des informations imprécises et incertaines, tout en se rapprochant des mécanismes de formalisation et de raisonnement humain. Les réseaux de neurones traitent des données expérimentales afin d'en extraire des informations. Dans les deux cas, des fonctions numériques non-linéaires peuvent être générées. Pour les problèmes de commande, la loi de régulation peut alors être établie sans utilisation nécessaire d'un modèle précis du système régulé. Des voies sont donc ouvertes pour obtenir différemment des lois de régulation pour des systèmes linéaires ou non-linéaires.

D'autre part, l'automatisation des procédés prend une place de plus en plus grande dans notre environnement technologique; les tâches se complexifient et se diversifient. Il apparaît donc clairement la nécessité de s'intéresser à de nouvelles approches de modélisation du monde réel. En effet, contrairement au physicien qui cherche à *converger vers la vérité objective* (modélisation la plus précise possible des phénomènes étudiés), l'ingénieur automaticien cherche à trouver le meilleur modèle possible pour un problème donné.

L'objectif général de notre travail était une évaluation des techniques floues et neuronales afin d'en déduire des possibilités de développement d'outils performants pour l'Automatique. Ces outils, devant être fiables et faciles à utiliser pour des industriels, peuvent éventuellement résulter de combinaisons d'approches. L'enjeu principal, pour que les méthodes utilisant des approches floues ou neuronales soient des outils pertinents pour l'automaticien, est d'établir le lien méthodologique entre les paramètres du module de commande et la fonction entrée-sortie numérique correspondante. Pour cela, deux directions principales de recherche peuvent être empruntées. La première consiste à prouver la fiabilité d'une approche par une voie théorique; ainsi, par exemple, se développent les travaux sur les équivalences ou sur la stabilité. La deuxième voie, plus méthodologique, correspond à la définition de procédures de réglage automatique afin d'atteindre des spécifications données en boucle fermée. Disposant de bancs d'essai sous forme d'un pendule inversé et de fours industriels, nos études ont été orientées vers des caractérisations expérimentales des apports des techniques étudiées. Les systèmes considérés ne possèdent pas de dynamiques complexes. Toutefois, l'utilisation de modèles du deuxième ordre non-linéaire permet de mettre au point des approches méthodologiques et de mener des études comparatives entre plusieurs structures de commande. Notre travail suit donc le deuxième axe de

recherche présenté plus haut, utilisant parfois des résultats du premier.

Plus précisément, nous avons réalisé: un régulateur flou (**RF**) pré-compilé sous forme d'une table de décision (pour un pendule inversé), un RF réglé avec un système flou d'auto-réglage (pour des fours industriels), un régulateur neuronal (**RN**) utilisant la rétropropagation (pour un pendule inversé) et des superviseurs flous (**SF**) de régulateurs PID académique ou industriel (pour des fours).

Nos apports principaux ont été:

- de proposer une technique pour réaliser une régulation floue efficace avec table de décision,
- de mettre en oeuvre et d'affiner une méthode d'auto-réglage d'un RF en partant d'une loi PI,
- d'analyser expérimentalement les causes d'amélioration des lois de régulation floue,
- de développer deux SF de PID en intégrant des garanties de stabilité et en analysant les relations *paramètres de conception - performances en boucle fermée*,
- de proposer une loi de régulation de type régime glissant construite automatiquement à partir d'un réseau de neurones.

Toutes nos méthodes de commande ont été avantageusement testées sur les bancs d'essai et comparées à des régulateurs à placement de pôles ou PID industriel fournis avec les systèmes.

Notre mémoire est structuré comme suit:

Le chapitre (1): permet de poser notre problème de façon détaillée. Après avoir justifié les nouveaux besoins pour l'automatisation avancée des procédés industriels (1.1), les techniques connexionistes et de l'Intelligence Artificielle sont introduites (1.2). Ensuite, les liens entre les nouveaux besoins et les nouvelles approches sont clarifiés (1.3). Finalement, les bancs d'essai expérimentaux, considérés dans le cadre de cette thèse, sont présentés (1.4).

Le chapitre (2): aborde des aspects théoriques (2.1), méthodologiques (2.2) et des problèmes de mise en oeuvre (2.3) de la régulation floue.

Le chapitre (3): présente des expériences sur le pendule inversé (3.1) et sur les fours (3.2). Il est ici question du régulateur flou de première génération, caractérisé par des réglages essentiellement heuristiques. Nous avons étudié des mises en oeuvre avec table de décision (ce qui nous a conduit à nous pencher sur des problèmes de précision) et des mises en oeuvre avec inférence en ligne (pour lesquelles nous avons analysé les non-linéarités générées par un RF couramment utilisé).

Le chapitre (4): traite du développement d'une technique de régulation adaptative neuronale. Après avoir fait des rappels succincts sur les réseaux neuronaux (4.1), nous avons détaillé la méthodologie de conception proposée d'un régulateur neuronal (4.2), puis exposé des résultats sur le pendule inversé destinés à valider la démarche de conception (4.3).

Le chapitre (5): traite de mise en oeuvre de modules de régulation adaptative par la combinaison de techniques d'Intelligence Artificielle et conventionnelles pour les fours. Après avoir présenté un état de l'art sur les *superviseurs flous* de régulateur PID (systèmes experts flous adaptant les paramètres du PID) (5.1), nous présentons nos structures de commande (5.2) et les applications sur les fours (5.3).

Le chapitre (6): concerne des régulateurs flous de deuxième génération. Après avoir décrit les approches pour introduire des non-linéarités dans la loi de régulation (6.1), la procédure d'auto-réglage de RF (6.2) et une extension à une régulation floue adaptée en ligne (6.3), nous proposons des résultats d'expériences (6.4).

Bonne lecture

Chapitre 1

Position du problème

Quels sont, de nos jours, les problèmes posés à l'automaticien ? Quels sont les outils dont il dispose ? Telles sont les deux questions principales qui sont abordées dans le présent chapitre, situant notre problématique. Nous cherchons à montrer que, face à la diversité des tâches à réaliser, les *nouvelles techniques* étudiées au cours de cette thèse (systèmes experts flous, réseaux neuromimétiques) s'avèrent très utiles pour compléter les outils de l'automaticien.

Dans le paragraphe (1.1) sont décrites les différentes tâches d'automatisation et les limitations des approches conventionnelles pour réaliser toutes ces tâches. Dans le paragraphe (1.2) les techniques connexionistes et les approches de l'Intelligence Artificielle (IA), dont le flou, sont introduites. Le paragraphe (1.3) présente des outils pour répondre de façon efficace aux problèmes posés. Enfin, le paragraphe (1.4) expose les objectifs du travail de thèse et les systèmes utilisés pour valider nos études.

1.1 Intégration de systèmes de contrôle-commande

1.1.1 L'automatisation avancée

Une tâche de base de l'Automaticien est de développer des régulateurs pour piloter des systèmes. En fonction des erreurs de sortie du système (qui sont les variables d'observation), l'organe de commande génère automatiquement les entrées du système (variables de commande) de façon à obtenir un comportement souhaité des sorties du processus.

L'augmentation des capacités de calcul des ordinateurs a conduit à l'intégration de nouvelles fonctions afin d'obtenir des résultats plus performants et plus fiables. Le problème global posé à l'automaticien est alors de gérer des tâches complexes en développant diverses fonctions automatisées, car il devient impossible pour un opérateur spécialisé d'avoir une vision globale du procédé. Les structures considérées sont dénommées Systèmes Numériques de Contrôle-Commande (SNCC).

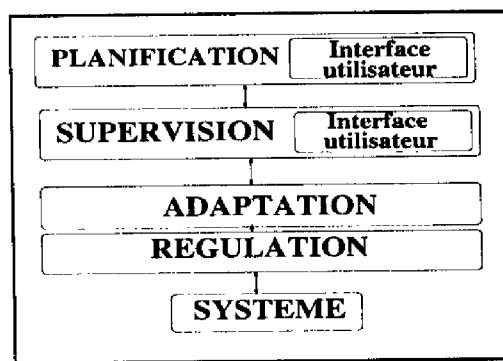


Figure 1.1 : Une architecture de SNCC.

Un exemple d'architecture est proposé sur la figure (1.1). La structuration en couches dépend de l'échelle de temps de chaque fonction: les niveaux supérieurs traitent des problèmes aux dynamiques les plus lentes. Les différents étages sont, en partant du bas vers le haut: les boucles de régulation, d'adaptation, de supervision et de planification.

Un nouveau concept caractérisant les SNCC est celui d'*intelligence*. Astrom parle de développer des contrôleurs intelligents ou autonomes, [AST,92]. Il ne s'agit pas d'espérer introduire dans un module technologique l'intelligence intégrale d'un opérateur humain avec toute sa richesse; la définition considérée est ici bien plus restrictive. L'intelligence du contrôleur peut être caractérisée par certaines capacités de raisonnement et d'apprentissage. *Sadiris* a caractérisé le *contrôle intelligent hiérarchique* par une précision qui augmente à mesure que l'intelligence diminue et inversement. Il s'agit donc de proposer des outils adaptés, fiables et simples à utiliser, pour aborder dans les meilleures conditions le problème de la commande des systèmes industriels.

A- Boucle de régulation

La boucle de régulation permet d'assurer le suivi de la référence en s'appuyant sur les informations des capteurs et en agissant sur les actionneurs. Les problèmes d'Automatique posés sont la régulation (les signaux de sortie du système doivent être

maintenus à un niveau constant quel que soit le niveau de perturbations), la poursuite (le signal de référence est variable)....

Les objectifs de la régulation ou de la poursuite sont d'atteindre un certain comportement dynamique en boucle fermée pour des domaines de fonctionnement donnés, tout en garantissant des contraintes globales sur la stabilité et la robustesse.

L'Automatique conventionnelle fournit une large famille de méthodes pour synthétiser des lois de commande, la plupart du temps, à partir d'un modèle suffisamment précis du système. En général, quatre étapes président à la conception de régulateurs: l'identification de la structure d'un modèle du système, l'estimation des paramètres correspondants, le choix d'un type de régulateur et le réglage de ses gains.

B- Boucle d'adaptation

La boucle d'adaptation module les valeurs des paramètres du régulateur, de façon à optimiser les résultats en présence de perturbations dans la boucle inférieure.

Trois approches principales ont été développées en fonction des grandeurs permettant de faire les adaptations, à savoir les paramètres du modèle du système, les erreurs de sortie du système ou les performances dynamiques atteintes. Dans le premier cas, le mécanisme d'adaptation peut être vu comme une automatisation des procédures de conception de modèles et de régulateurs. Les gains du régulateur sont adaptés en résolvant un problème de conception, à partir d'un modèle du système, dont les paramètres sont réactualisés par une estimation récursive, voir la figure (1.2). Il est question de régulateurs auto-réglants (*self tuning*: **STAC**) ou méthode adaptative indirecte, [AST,89]. Dans le second cas, les spécifications en boucle fermée sont exprimées par un modèle de référence. Il est question de régulateurs avec modèle de référence (**MRAC**) ou méthode de régulation adaptative directe, voir la figure (1.3). Les actions de réglage sont menées en fonction de l'écart quadratique entre les sorties du système et du modèle (e), grâce à une technique d'optimisation de type gradient. Dans le troisième cas, l'adaptation est basée sur une méthode de type reconnaissance de situation (*pattern recognition*: **PRAC**), [AST,89]. Les paramètres du régulateur sont modifiés à partir des caractéristiques de la réponse transitoire avec une démarche heuristique, voir la figure (1.4).

Des *applications* des approches adaptatives sont les techniques d'auto-réglage, [AST,89]. Elles sont une version automatisée du réglage des régulateurs. Une phase expérimentale avec des signaux tests et des règles d'adaptation permet de déduire les valeurs des paramètres du régulateur. L'auto-réglage d'un régulateur PID peut servir d'exemple. Les formules de réglage de *Ziegler & Nichols* sont seulement liées à la connaissance de l'ordre du système, de la période critique T_u des oscillations limites et du gain critique G_u du système bouclé. Dans ce cas, un modèle très précis du système n'est pas indispensable. Les informations nécessaires sont fournies, par exemple, expérimentalement avec la méthode du relais, [AST,92].

Une autre technique adaptative couramment utilisée est la planification des gains (*gain scheduling*) qui consiste à fournir au régulateur les gains correspondant aux nouvelles conditions opérationnelles, [AST,89]. La technique peut être mise en place

de plusieurs manières: soit un effort important de conception est mené hors ligne pour un large domaine de conditions opérationnelles, soit les informations nécessaires sont stockées en ligne à l'aide d'un superviseur. Parfois, ce type de module n'est pas considéré comme un module adaptatif, car l'adaptation avec les informations établies préalablement peut être faite de manière anticipative sans aucune rétroaction venant du système, [AST,89].

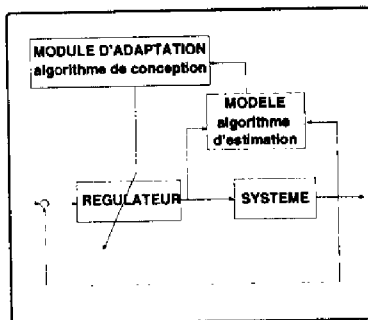


Figure 1.2 : STAC.

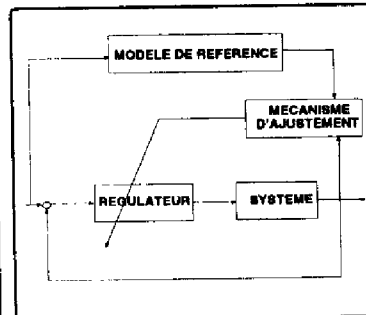


Figure 1.3 : MRAC.

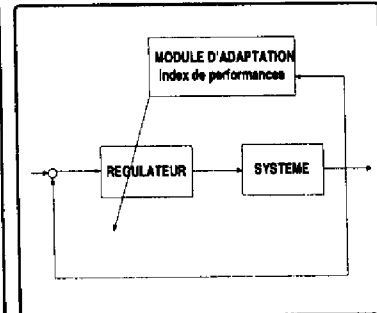


Figure 1.4 : PRAC.

C- Boucle de supervision

La boucle de supervision englobe l'ensemble des mécanismes à mettre en oeuvre pour conduire un système dans les conditions de perturbation les plus générales, y compris en situation de défaillances. Le superviseur décide, parfois en association avec l'opérateur, quand et comment agir sur la boucle de régulation pour atteindre les objectifs fixés. Plusieurs sortes de fonctions peuvent être gérées par un superviseur. Un certain nombre d'entre elles ont pour rôle de rendre compte à l'utilisateur de situations, afin de l'aider à analyser et à décider; les autres fonctions ont pour but d'agir de façon autonome.

Dans la première catégorie, il est possible d'inclure les aspects détection et diagnostic de fautes. Les fonctions d'aide à la décision incluent également des modules de présentation de l'évolution de certaines données en ligne (*process monitoring*), d'explication d'informations utiles à l'utilisateur sur les actions automatiques menées, de stockage d'historiques permettant des traitements de données relatives à la boucle bas niveau.

Des fonctions du superviseur exécutées sans intervention d'un opérateur, dans le cas d'évènements prioritaires, sont, par exemple, des fonctions de protection automatique en cas d'alarme, ou de réparation de fautes. Il est aussi possible de citer des procédures d'orchestration de différents algorithmes pour atteindre des objectifs de régulation spécifiés par l'étage supérieur. Des exemples sont: la décision de procéder à un nouveau réglage du régulateur en place ou à une nouvelle estimation de paramètres pour le modèle, le choix d'un type de modèle (ordre....) ou de régulateur

(Proportionnel-Intégral-Dérivé (**PID**), Proportionnel-Dérivé (**PD**), **PI**, etc....). Enfin, peuvent être intégrées au superviseur des fonctions de mémorisation d'un réglage pour un contexte donné, ou de définition d'une sous-consigne.

D- Boucle de planification

La boucle de planification permet de définir de nouveaux objectifs et d'élaborer des stratégies de régulation. Elle possède une interface utilisateur afin d'intégrer l'Homme dans le processus d'automatisation. L'automatisation avancée des procédés peut permettre de répartir les tâches entre l'Homme et la machine de façon à laisser systématiquement aux ordinateurs les travaux répétitifs et pénibles et à donner à l'utilisateur une place valorisante, [MAR,83]. Le fait de permettre aux opérateurs d'exercer leurs compétences régulièrement, d'être mieux assistés avec des informations temps réel utiles pour une collaboration effective, [GEN,94], entraîne une augmentation de la productivité par amélioration des conditions de travail ([EFS,92]) et une diminution des problèmes de sûreté de fonctionnement ([ROD,92]).

1.1.2 Limitations des approches conventionnelles

La plupart des approches conventionnelles utilisent parfois des hypothèses trop restrictives, comme, par exemple, la nécessité d'un modèle mathématique précis du système régulé. Ainsi, si la structure du modèle choisie pour un problème de régulation adaptative de type *self-tuning* n'est pas suffisamment proche de la réalité, la méthode d'adaptation ne fonctionnera pas [AST,86]. Tous les problèmes de régulation ne peuvent donc pas être résolus efficacement, en particulier ceux pour lesquels la modélisation analytique du procédé est difficile, voire impossible; or, ces cas de figure se rencontrent pour beaucoup de systèmes industriels, [ROD,92]. De façon générale, les méthodes conventionnelles sont précises; toutefois, le codage procédural ne permet pas beaucoup de généralisation, il est spécifique à une application donnée et hautement spécialisé. Des limites apparaissent donc clairement pour l'intégration des fonctions d'automatisation avancée avec seulement des fonctions mathématiques.

Le deuxième type de limitations est le nombre réduit de degrés de liberté de certaines lois de régulation, ce qui les rend surtout efficaces pour les systèmes linéaires, [VER,92]. Un exemple est le régulateur PID, qui est très largement utilisé pour les applications industrielles (à 90%), [TZA,90]. Il possède l'avantage d'être facile à utiliser et sûr pour des conditions de fonctionnement nominales; mais, en contrepartie, il possède la limitation majeure d'être réglé sous-optimalement dans la plupart des autres conditions de fonctionnement (telles des variations importantes de facteurs de la boucle de régulation -ces facteurs pouvant être des dynamiques non modélisées et très influentes du système, du bruit ou le niveau de consigne-), [LIT,91] [OLI,91] [NAU,90] [TZA,90]. Des ajustements supplémentaires sont alors souvent nécessaires dans les contextes nouveaux cités précédemment, mais aussi, parfois, après la phase

de réglage des gains du PID, [NAU,90]. En effet, les méthodes de réglage couramment utilisées, comme les formules de *Ziegler & Nichols*, conduisent à des résultats souvent inacceptables en pratique. C'est pourquoi, dans l'industrie, plus de 50% des régulations sont menées en mode manuel car les opérateurs sont insatisfaits des performances atteintes après l'application de procédures d'auto-réglage, [LIT,91].

Certaines méthodes conventionnelles sont parfois difficiles à utiliser et trouvent peu de succès auprès des industriels, qui ne disposent pas toujours des compétences nécessaires. Il s'agit surtout des techniques de régulation des systèmes non linéaires; ces régulateurs, possédant de nombreux paramètres, sont flexibles. Mais, pour les régulateurs conventionnels, flexibilité et transparence (clarté des liens entre les paramètres du régulateur et les performances en boucle fermée) s'opposent souvent.

Enfin, la dernière remarque à propos des limitations rencontrées à ce jour dans le domaine des approches mathématiques, concerne certains aspects théoriques. Pour les modules adaptatifs, par exemple, beaucoup de temps a été nécessaire pour mettre au point leur conception et comprendre leur comportement; toutefois, un nombre non négligeable de problèmes théoriques subsistent encore, [AST,86]. Tout d'abord, le domaine ne peut pas être considéré comme mature car des approches systématiques de conception manquent encore, [AST,89]. De plus, concernant les preuves de stabilité, aucune preuve n'existe dans le cas des méthodes de planification de gain. Pour les modules purement adaptatifs, des preuves globales ont été apportées mais sous des hypothèses très restrictives; des résultats locaux sont souvent utilisés, [AST,89].

Les différents niveaux des SNCC réalisent donc des fonctions variées qui appellent des méthodes de conception variées. Les approches conventionnelles possédant certaines limitations, il est intéressant de se tourner vers d'autres approches, telles que les techniques appelées "non-conventionnelles". Ces outils regroupent les domaines de l'IA -incluant les techniques floues-, les approches neuromimétiques et les méthodes de raisonnement probabiliste -incluant les algorithmes génétiques-.

1.2 Les techniques connexionistes et floues

1.2.1 Les réseaux de neurones

A- Principe de l'approche

Un réseau de neurones est un module technologique organisé en couches. Il peut être représenté par un graphe dont les arêtes sont des voies de transmission des données et les sommets des processeurs élémentaires de données numériques. Ces derniers sont appelés neurones formels; ils réalisent en général des fonctions d'activation non linéaires. A chaque voie de transmission entre deux neurones est associé un poids

synaptique. Cette structure conduit à une représentation interne de l'information. En effet, la connaissance est contenue dans les poids, donc dans une famille de vecteurs réels, qui peut être représentée sous la forme d'une matrice de connexions synaptiques.

Une propriété caractéristique des réseaux de neurones est leur capacité d'*apprentissage*. Le problème posé est de déterminer la matrice des poids synaptiques permettant au réseau d'avoir un certain comportement. Deux phases d'utilisation du réseau peuvent être différenciées: une phase d'apprentissage et une phase d'utilisation. La première consiste à faire converger un algorithme itératif vers un point stable. Plusieurs techniques sont utilisables suivant que le comportement désiré est connu ou non: l'apprentissage est alors soit supervisé, soit non supervisé. Dans le premier cas, les sorties attendues du réseau pour un vecteur d'entrées donné sont connues, elles sont donc utilisées pour faire converger les paramètres de la structure vers leur valeur optimale. Dans le second cas, les valeurs désirées des sorties ne peuvent pas être formulées, la procédure d'apprentissage doit donc converger de façon non guidée. Le choix d'une technique d'apprentissage dépend du type de réseau utilisé (architecture, type d'éléments de la structure....) et de la fonction attribuée à ce réseau (classification, identification....)¹.

B- Potentialités

Une propriété fondamentale des réseaux de neurones avec apprentissage supervisé est celle d'approximation universelle parcimonieuse. Etant donnée une fonction non linéaire continue, il existe un réseau de neurones qui peut approximer cette fonction pour un domaine donné des entrées; pour plus de détails, voir le paragraphe (4.2.4) du chapitre (4). Dans le cas où le nombre d'entrées augmente, l'approximation polynomiale avec une méthode des moindres carrés conduit à une explosion du nombre de coefficients et d'exemples nécessaires. Par contre, le réseau neuronal nécessite moins de paramètres et d'exemples pour une approximation équivalente.

Les réseaux neuronaux structurent la connaissance contenue dans des données expérimentales, de façon transparente pour l'utilisateur. Les cas typiques d'utilisation de ces outils sont des situations pour lesquelles beaucoup d'exemples numériques, caractéristiques du comportement attendu, existent et très peu de connaissance sur la façon de résoudre le problème est disponible. L'architecture du réseau peut être définie de façon *aveugle*, mais, bien sûr, toute connaissance initiale disponible mérite d'être intégrée dans la définition de l'architecture afin d'augmenter l'efficacité du travail de conception.

Une conséquence intéressante de la propriété d'approximation universelle est de fournir de nouveaux outils pour les problèmes de modélisation; en effet, une possibilité est donnée d'identifier des systèmes non linéaires quelconques, même si la structure du modèle est complètement inconnu.

Les réseaux neuronaux sont potentiellement très intéressants pour réguler des systèmes, pour deux aspects principaux. Le premier aspect est leur capacité à manipuler des données entrées/sorties du système afin de construire empiriquement des lois de

¹Nous nous limitons aux réseaux de neurones statiques multicouches (*feedforward*).

régulation, sans qu'aucune connaissance d'un modèle analytique précis du système ne soit nécessaire. De plus, une fois que l'apprentissage d'une relation entrée/sortie a été réalisé, le temps de génération d'une réponse pendant la phase d'utilisation est rapide, particulièrement avec des composants matériels dédiés. Le deuxième aspect est leur caractère adaptatif; lors de la phase d'apprentissage, une adaptation continue des paramètres du réseau est menée en ligne, d'où une robustesse potentielle très grande dans le cas de conditions de régulation perturbées.

C- Des limitations

Bien sûr, les réseaux neuronaux ne sont pas les *outils miracles* et un certain nombre de leurs qualités possèdent des contreparties. La stratégie boîte noire est attirante pour construire de la connaissance à partir d'une expertise très faible et de données numériques mais, d'un autre côté, elle n'est pas transparente. Il est difficile d'analyser la connaissance codée de façon distribuée dans un réseau. D'autre part, des problèmes de convergence se posent parfois avec différentes techniques d'apprentissage. La garantie de converger vers l'optimum global du problème n'est pas toujours donnée et les techniques qui permettent de s'assurer d'une bonne convergence conduisent à des recherches d'optima très longues. Enfin, beaucoup de choix heuristiques sont nécessaires pour fixer les paramètres principaux des réseaux neuronaux.

1.2.2 Les techniques d'IA, et plus particulièrement le flou

A- Potentialités

a- Type de connaissance modélisée et mode de traitement de l'information:
Les informations heuristiques doivent parfois être exploitées. En effet, les opérateurs spécialisés dans la conduite de certains systèmes sont souvent essentiels pour pallier les limitations des algorithmes de régulation, par exemple dans le cas de processus difficilement modélisables. Or, la connaissance experte est facilement modélisable par des règles fondées sur l'intuition et l'expérience, [BAB,92]; par exemple, on dérive facilement un ensemble de règles d'adaptation, à partir des actions menées par des opérateurs adaptant les régulateurs PID.

De plus, le formalisme flou est encore plus proche de la perception humaine de la réalité que le formalisme linguistique standard. Cette qualité résulte de la facilité d'intégration, dans les systèmes flous, des notions subjectives (ex.: *confort*), graduelles (pour les problèmes de seuillage) et incomplètes (imprécises ou incertaines, [DUB,91]). Les ensembles flous permettent une certaine simplicité de codage de tels concepts: l'algorithme déclaratif manipule les concepts linguistiques, alors qu'à un niveau plus bas se définit l'interface numérique-symbolique ajustable. Il suffit de modifier la définition numérique des ensembles flous pour modifier la signification subjective des notions manipulées. Ceci entraîne une facilité d'utilisation des modules flous.

Une autre raison, rendant les systèmes experts très attractifs pour modéliser la connaissance humaine, est l'exécution des algorithmes déclaratifs grâce à un mécanisme proche du raisonnement humain. La résolution de problèmes se fait par un

processus d'inférence qui permet de déduire de la connaissance.

Le mode de traitement des règles floues est plus proche encore de la démarche de raisonnement humain, que le mécanisme utilisé avec le formalisme linguistique classique. Le mécanisme d'inférence flou permet de développer un raisonnement approché.

b- Atouts de la régulation floue: Les systèmes flous constituent une nouvelle approche pour modéliser et réguler des systèmes. L'objectif n'est pas de remettre en cause la *méthodologie du modèle*, mais d'en élargir le champ, [DUB,90].

Ils permettent d'augmenter la capacité de modélisation et de régulation, en s'appliquant à une classe plus large de systèmes que ceux traitables avec les approches algorithmiques. Ils peuvent, par exemple, permettre de modéliser ou réguler des systèmes dont le modèle analytique est inconnu. Les situations où leur utilisation est recommandée sont celles où beaucoup de connaissance experte existe et peu d'exemples numériques typiques du comportement attendu sont disponibles. L'approche floue est plus naturelle pour la description immédiate des phénomènes physiques.

Dans les cas de modèles connus, les régulateurs flous sont aussi parfois une alternative technologique. En effet, le formalisme et le mode de raisonnement utilisés sont facilement compréhensibles par le concepteur et l'utilisateur, [OLI,91]; les règles floues s'avèrent plus facilement exploitables que des algorithmes procéduraux complexes ou même que des règles classiques. Il en découle des coûts de conception et d'utilisation inférieurs à ceux des techniques conventionnelles.

De plus, les nombreux paramètres du RF entraînent une flexibilité intéressante pour générer des lois de régulation non linéaires. Enfin, les RF possèdent une certaine transparence, car les liens entre les valeurs de leurs paramètres principaux et les effets sur les performances sont intuitivement facilement compréhensibles. Il en résulte des temps de développement courts pour aboutir à un réglage satisfaisant. L'obtention d'un réglage optimal est un problème plus compliqué. Ce point motive des recherches sur les équivalences entre formalisme flou et fonctions analytiques, ainsi que sur les procédures de réglage automatique de l'interpolateur non linéaire qu'est le RF.

B- Des critiques

Ces dernières années, de sérieuses réticences d'une partie de la communauté de l'Automatique ont vu le jour, face à l'émergence de nouveaux formalismes de résolution de problèmes, [DUB,90]. Un certain nombre des critiques avancées et d'éléments de modération sont rappelés ici.

Une première critique consiste à affirmer que *la logique floue est floue*. Cette remarque tend à remettre en cause les fondements mathématiques de la logique floue; pourtant l'exploitation de modèles basés sur le flou donne des résultats précis et déterministes. Ainsi, *"la théorie des probabilités peut modéliser le hasard, mais ce n'est pas pour autant que la validation de ses théorèmes est elle-même un processus aléatoire"*, [DUB,90]. De plus, il est également reproché à la logique floue d'être une approche qui revient à considérer que le scientifique est condamné à ne modéliser la réalité qu'au degré de perception de l'homme. Or *"réduire la philosophie des ensembles flous à la*

constatation désespérée que seule la théorie du vague permet de modéliser l'univers revient à considérer que les probabilistes se réfugient dans la constatation désespérée que seule une théorie du hasard permet de modéliser l'univers", [DUB,90]. Il faut considérer la logique floue et les outils dérivés comme de nouvelles approches pour modéliser les phénomènes physiques. En outre, la dimension mathématique sous-jacente existe, mais les mécanismes de génération de fonctions analytiques précises n'étaient pas compris dans un premier temps; *"l'expérience a devancé la théorie comme c'est souvent le cas dans l'histoire des sciences pour l'ingénieur"*, [DUB,90].

Il est souvent reproché aux concepteurs de régulateurs flous de réaliser leurs réglages par des approches trop heuristiques, qui reposent sur des essais expérimentaux. La première génération de RF correspond, en effet, à ce cas de figure. Mais la deuxième génération de RF est caractérisée par le développement de procédures de réglage automatique, qui demeurent un des axes de recherche essentiels. De plus, les études théoriques permettant de formaliser les équivalences entre des lois de régulation floue et des fonctions analytiques se poursuivent, [FOU,92].

Une autre critique insiste sur la difficulté d'obtenir de la connaissance experte. Trois sources d'information permettent de construire un module linguistique: des observations qualitatives sur le système physique considéré, les descriptions trouvées dans des documents relatifs au procédé ou l'interrogation d'un expert humain spécialiste du système. Cette dernière étape n'est pas toujours triviale, mais elle est parfois nécessaire en l'absence d'autres sources de connaissance. Le travail d'extraction, ou plutôt de formalisation, de la connaissance experte est un travail scientifique et psychologique. En effet, pour reconstituer l'information, il faut faire appel aux sciences de la psychologie cognitive, par l'utilisation de techniques d'interrogation de personnes (discussions, questionnaires,...), et aux sciences plus technologiques, pour reproduire les raisonnements logiques utilisés par l'expert avec des méthodes de recherche dans des arbres, [GEN,94].

La critique, insistant sur les problèmes de validation de la stabilité avec les RF, est de moins en moins d'actualité. En effet, les recherches effectuées dans ce domaine permettent d'offrir, dans certains cas, des méthodes de conception garantissant des marges de stabilité et des propriétés de robustesse, [GAR,94]. Dans d'autres cas, des analyses a posteriori permettent de tester la stabilité en boucle fermée, [RUG,94]. Deux directions de recherche sont suivies: le travail sur les équivalences, permettant d'utiliser les techniques d'analyse de stabilité conventionnelle, et la recherche de nouvelles méthodes d'analyse de stabilité dans le domaine linguistique. Le paragraphe (2.3.1) du chapitre (2) donne quelques précisions supplémentaires.

Une dernière critique importante est que les systèmes complexes ne sont pas réglables avec des RF. Il est parfois reproché aux concepteurs de régulateurs flous de se contenter de considérer des systèmes avec des dynamiques très simples, ce qui explique leurs bons résultats. Les applications sur des systèmes simples sont nécessaires pour valider les méthodologies de conception de régulateurs flous, pour évaluer leurs apports, éventuellement via des études comparatives. De plus, des applications concluantes ont été réalisées sur des systèmes complexes industriels. Il est possible de citer une commande floue de fours à ciment développé par *Holmblad* en 1982 qui a

permis de développer une méthode de conception plus rapide qu'avec des techniques conventionnelles et d'obtenir de meilleures performances. D'autres exemples sont des contrôles d'unités de production chimique, la commande d'un hélicoptère sans pilote (*Sugeno, 1991*), la régulation prédictive floue du métro de Sendai (*Hitachi's System Dev lab, Kawasaki, 1986*) conduisant à des résultats très intéressants de précision d'arrêt, d'économie d'énergie et de confort.

Toutefois, des recherches théoriques restent nécessaires pour appréhender efficacement la régulation de systèmes complexes.

1.3 Le bon outil pour un problème d'automatisation

1.3.1 Approche méthodologique

L'automatisation peut donc être vue comme l'intersection entre plusieurs domaines technologiques, voir la figure (1.5). Chaque ensemble est plus approprié pour certaines tâches et apporte un peu plus de flexibilité aux SNCC. L'IA fournit des modules de raisonnement, les réseaux neuromimétiques des outils de traitements de données numériques, les techniques conventionnelles des méthodes analytiques le plus souvent validées théoriquement. D'autres méthodes, non traitées ici, existent, telles les algorithmes génétiques, la modélisation qualitative....

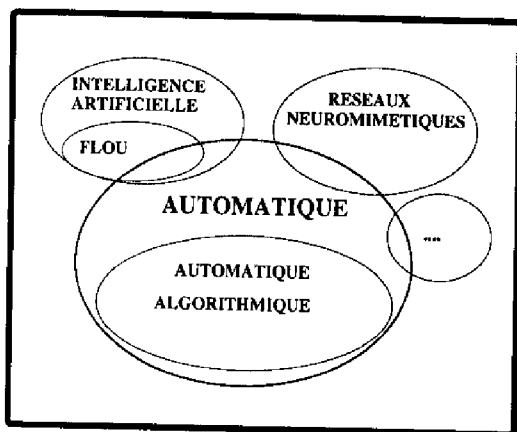


Figure 1.5 : Des outils pour l'Automatique.

Une approche méthodologique est nécessaire pour choisir le bon outil pour un problème donné. Il s'agit d'utiliser un niveau de détail cohérent avec la nature de la tâche à remplir, les caractéristiques de la connaissance disponible et la précision requise, [LEI,92]. La notion de granularité du problème à résoudre est fondamentale. En effet, si la complexité d'un système augmente, la possibilité d'obtenir une description précise diminue (*principe d'incompatibilité de Zadeh*).

Le problème est d'exploiter la tolérance à l'imprécision des phénomènes physiques, pour dériver des méthodes de calcul qui conduisent à une solution acceptable, à faible coût, car la haute précision entraîne un coût élevé, [ZAD,92]. Ainsi, bien qu'en physique, l'objectif est de converger vers le meilleur modèle de la réalité, en automatique, un modèle est correct s'il permet de résoudre au mieux le problème posé.

1.3.2 Des combinaisons de techniques

Les techniques considérées ici ont des domaines d'application différents et possèdent aussi des complémentarités intéressantes, [ZAD,92]. Des coopérations étroites entre les différentes approches permettent de générer des systèmes avec des degrés d'intelligence non triviaux, [AST,86].

La combinaison entre des démarches analytiques dans la boucle bas niveau et des systèmes experts placés au-dessus permet d'allier des expressions mathématiques bien maîtrisées avec des mécanismes de raisonnement manipulant de la connaissance heuristique. Cette approche peut conduire à améliorer les résultats de systèmes conventionnels déjà largement mis en oeuvre, comme les régulateurs PID, [OLI,91]. Un superviseur flou de régulateur PID allie les garanties sécurisantes de stabilité de l'algorithme PID, dans des conditions nominales, avec l'adaptabilité d'un formalisme linguistique flou à des situations nouvelles, [NAU,90].

D'autre part, les systèmes experts flous ou les réseaux de neurones offrent de nouvelles méthodologies de saisie de la connaissance et de modélisation de lois non linéaires et sont, à ce titre, également adaptés pour la régulation bas niveau. Un type de combinaison intéressant peut alors être l'association parallèle avec un régulateur linéaire, [KET,93] [KET,95]. Une autre association pertinente concerne un modèle flou ou neuronal pour un régulateur conventionnel, [DEG,91].

Dans le cas de régulations floues, des méthodes de réglage automatique sont souhaitables. Elles peuvent être mises en place grâce à des outils neuronaux, complétant la connaissance experte à partir de données expérimentales. Le réseau, pré-codé avec la connaissance linguistique disponible, peut, par exemple, apprendre les valeurs des paramètres du régulateur flou, qui sont ensuite recodées dans le module flou. Le réseau neuronal peut également traiter des données numériques pour en déduire certains paramètres du régulateur flou, tels les règles, [CUL,95]. D'autre part, les méthodes de réglage de régulateurs flous peuvent également reposer sur des formulations mathématiques, comme des techniques d'optimisation.

1.4 Objectifs du travail de thèse

Notre travail a pour but d'évaluer les apports des approches symboliques et connexionnistes à l'Automatisation des procédés, à partir d'analyses expérimentales. Ces études ont nécessité des comparaisons avec des outils de régulation conventionnelle. Elles ont conduit à des réflexions sur les stratégies de conception de lois de régulation floue et neuronale, afin d'assurer certaines performances en boucle fermée.

La première étape de notre travail a consisté à étudier des régulateurs bas niveau, afin de se familiariser avec les techniques considérées et d'en dégager les avantages et inconvénients. La deuxième étape correspond à des mises en oeuvre de commandes de type adaptatives.

Dans le cas des régulateurs neuronaux, nous nous sommes attachée à établir un lien entre certains paramètres de conception du réseau neuromimétique et les performances dynamiques atteintes en boucle fermée. Dans le cas des régulateurs flous, après avoir préalablement manipulé des régulateurs flous de la première génération, cette voie de recherche nous a conduite à établir des procédures de réglage automatique, donc à nous orienter vers les RF de deuxième génération. L'exploration d'outils pour l'Automatique, combinant différentes techniques, nous a amenée à associer un module expert flou à un algorithme mathématique très largement répandu, qu'est le régulateur PID.

Les expériences d'évaluation des différents modules de commande réalisés ont été menées sur l'un ou l'autre des deux systèmes suivants. Le premier support expérimental est un pendule inversé utilisé en simulation ou en application temps réel. Le deuxième type de banc d'essai est un four industriel, sous forme de modèles ou d'un simulateur temps réel. Dans tous les cas, les modèles des systèmes considérés ont été nécessaires, non pas pour concevoir les lois de régulation, mais pour les essais en simulations.

1.4.1 Le pendule inversé

Le choix d'un pendule inversé comme banc d'essai est justifié par les raisons suivantes. Tout d'abord, la position d'équilibre instable visée (l'objectif de la stratégie de régulation est d'amener puis de maintenir la tige du pendule en position verticale haute) et la dynamique non linéaire rendent le système intéressant pour évaluer la qualité des commandes mises en place. De plus, ce système est communément utilisé, voir [MOR,76] pour une application avec des techniques de régulation conventionnelles et [YAM,89] pour une mise en oeuvre d'un régulateur flou. Des comparaisons, avec les résultats d'autres équipes de recherche, sont alors possibles. Enfin, l'étude de la dynamique du pendule inversé présente une utilité pratique, puisqu'elle est proche de celle d'une fusée en phase de lancement, ou de certains éléments de robots mobiles.

A- Modèle mécanique établi

Le pendule inversé considéré pour les simulations est représenté sur la figure (1.6). Les quatre variables principales sont: la position angulaire de la tige θ , la vitesse angulaire $\dot{\theta}$, la position horizontale du chariot y et la vitesse de déplacement du chariot \dot{y} .

Nous avons utilisé les équations fondamentales de la mécanique pour dériver le modèle du pendule inversé. Il s'agit des théorèmes des forces extérieures, voir l'équation (1.1) et du moment cinétique, voir la formule (1.2). Ils sont appliqués au deux sous-systèmes chariot et tige de la figure (1.7), de masses M pour le chariot, m pour la tige et L représente la demie-longueur de la tige. Les mouvements sont exprimés par rapport au repère galiléen R_{a_0} . Dans la première équation, G représente le centre de gravité du sous-système considéré, $\vec{\Gamma}_G$ est le vecteur accélération en G et \vec{F}_{ext_G} une force extérieure appliquée en G . Dans la deuxième équation, $\vec{M}_{om_{F_{ext_G}}}$ est

le moment d'une force extérieure et $\vec{\sigma}_G$ est le vecteur moment cinétique; ces vecteurs sont appliqués aux centres de gravité des sous-systèmes, en l'absence d'un point fixe dans le repère R_{a_0} .

$$\left(\sum \vec{F}_{extG}\right)_{R_{a_0}} = M (\vec{\Gamma}_G)_{R_{a_0}} \quad (1.1)$$

$$\left(\sum \vec{M}_{om_{F_{extG}}}\right)_{R_{a_0}} = \left(\frac{d\vec{\sigma}_G}{dt}\right)_{R_{a_0}} \quad (1.2)$$

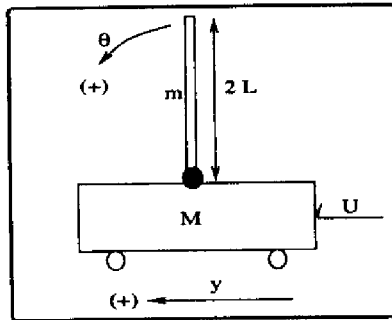


Figure 1.6 : Pendule inversé.

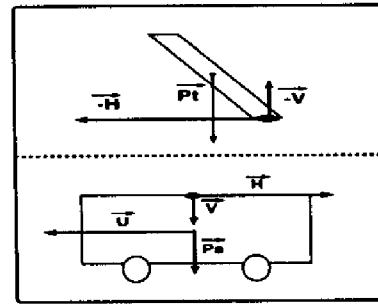


Figure 1.7 : Etude mécanique.

Les équations mécaniques sont données en (1.3). La première équation correspond à l'application du théorème des forces extérieures sur le sous-système support. Les deux suivantes résultent de l'utilisation du même théorème sur la tige. La dernière expression résulte de l'application à la tige du théorème du moment cinétique.

$$\begin{cases} M \ddot{y} = U - H \\ H = m \ddot{y} + mL [\ddot{\theta} \cos(\theta) - \dot{\theta}^2 \sin(\theta)] \\ m g - V = mL [\ddot{\theta} \sin(\theta) + \dot{\theta}^2 \cos(\theta)] \\ L V \sin(\theta) - L H \cos(\theta) = (J + mL^2) \ddot{\theta} \\ J = \frac{mL^2}{3} \end{cases} \quad (1.3)$$

Plusieurs cas de figure se présentent, en fonction du nombre de variables de sortie considérées (θ , $\dot{\theta}$ ou θ , $\dot{\theta}$, y et \dot{y}), et des forces négligées (force de collage, frottements secs et frottements visqueux). Nous avons utilisé, pour les simulations, les modèles correspondant à un système sans frottement (modèle non physique, mais souvent rencontré dans la littérature).

B- Modèles différentiels non linéaires déduits

Le système est modélisé par un système de (2) ou (4) équations différentielles non linéaires du premier ordre (1.4) (utilisées seulement pour la régulation de θ) et (1.5). Ces équations non linéaires sont intégrées numériquement par une méthode de *Runge Kutta*.

$$\begin{cases} \dot{X}_1 = \dot{\theta} = X_2 \\ \dot{X}_2 = \ddot{\theta} = \frac{\frac{Lmg}{I} \sin(X_1) - \frac{mL}{(m+M)I} \cos(X_1) U - \frac{m^2 L^2}{(M+m)I} \cos(X_1) \sin(X_1) X_2^2}{1 - \frac{m^2 L^2}{(M+m)I} \cos^2(X_1)} \end{cases} \quad (1.4)$$

$$\begin{cases} \dot{X}_3 = \dot{y} = X_4 \\ \dot{X}_4 = \ddot{y} = \frac{\frac{1}{M+m} U + \frac{mL}{m+M} \sin(X_1) X_2^2 - \frac{m^2 L^2}{(M+m)I} g \sin(X_1) \cos(X_1)}{1 - \frac{m^2 L^2}{(M+m)I} \cos^2(X_1)} \end{cases} \quad (1.5)$$

$$\text{avec: } I = J + mL^2 \quad (1.6)$$

C- Modèle linéaire dérivé

Le modèle linéaire, nécessaire pour des comparaisons avec des résultats de la littérature, a été obtenu pour des angles appartenant à $[-10^\circ, 10^\circ]$. Les approximations effectuées sont $\sin(\theta) \approx \theta$, $\cos(\theta) \approx 1$ et $\frac{\partial[\dot{\theta} \cos(\theta)]}{\partial \dot{\theta}} \approx \ddot{\theta}$. La formule (1.7) décrit la modélisation de Laplace (où θ_{init} représente la condition angulaire initiale). Les équations récurrentes du modèle discret, obtenues avec un bloqueur d'ordre zéro, sont données en (1.8) (où δ représente le signal de Dirhac et T_e la période d'échantillonnage).

$$\begin{cases} \theta(p) = \frac{B u(p) + p \theta_{init}}{p^2 - A} & \text{avec } A = \frac{3g(M+m)}{L(7M+4m)} ; B = -\frac{A}{g(M+m)} \\ y(p) = \frac{C \theta(p) + D u(p)}{p^2} & \text{avec } C = B mgL ; D = -B \frac{7L}{3} \end{cases} \quad (1.7)$$

$$\begin{cases} \theta(k+1) = \theta(k)[e^{\sqrt{A}T_e} + e^{-\sqrt{A}T_e}] - \theta(k-1) + u^\theta[u(k) + u(k-1)] + \delta^\theta[\delta(k) + \delta(k-1)] \\ \dot{\theta}(k+1) = \dot{\theta}(k)[e^{\sqrt{A}T_e} + e^{-\sqrt{A}T_e}] - \dot{\theta}(k-1) + u^{\dot{\theta}}[u(k) + u(k-1)] + \frac{\theta_{init}}{2}[\delta(k) + \delta(k-1)] \\ y(k+1) = 2y(k) - y(k-1) + \frac{T_e^2}{2}[C \theta(k) + C \theta(k-1) + D u(k) + D u(k-1)] \\ \dot{y}(k+1) = \dot{y}(k) + T_e C \theta(k) + T_e D u(k) \\ \text{avec } u^\theta = \frac{B(e^{\sqrt{A}T_e} + e^{-\sqrt{A}T_e} - 2)}{2A} ; \delta^\theta = \frac{\theta_{init}(e^{\sqrt{A}T_e} - e^{-\sqrt{A}T_e})}{2\sqrt{A}} ; u^{\dot{\theta}} = \frac{B}{\theta_{init}} \delta^\theta \end{cases} \quad (1.8)$$

D- Système réel

Le système réel se différencie des modèles établis précédemment par plusieurs aspects. Tout d'abord, les frottements modifient le comportement dynamique. De plus, les moments d'inertie sont légèrement différents (la tige est formée de deux cylindres de diamètres différents bout-à-bout dans le cas du système réel). Enfin, la commande envoyée au dispositif expérimental est une tension qui permet de commander un moteur à courant continu.

Le schéma du dispositif expérimental est fourni en figure (1.8). Un chariot, surmonté d'une tige, se déplace horizontalement sur une courroie actionnée par un moteur à courant continu. L'ordinateur pilote le moteur à courant continu par l'intermédiaire d'une carte convertisseur et d'une unité actionneur, comprenant un étage d'adaptation des tensions des signaux et un étage de puissance.

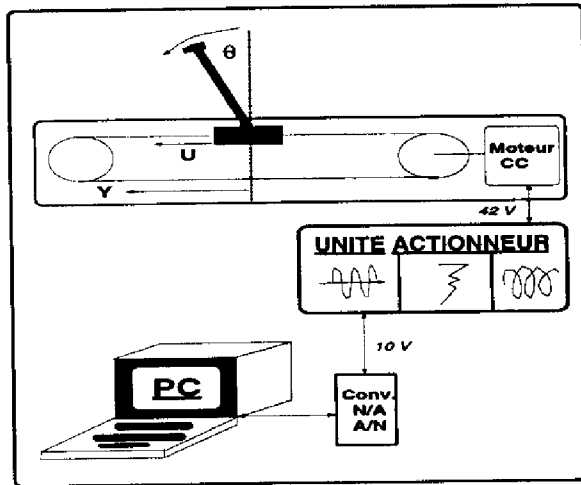


Figure 1.8 : Le banc d'essai du pendule inversé.

Sur la figure (1.8), Y définit la position horizontale par rapport au centre du rail, sa valeur maximale est de 66cm; U représente la force horizontale motrice, elle atteint la valeur limite de 120N; θ symbolise l'angle de la tige par rapport à l'axe vertical, il varie entre les bornes $[-\pi, \pi]$. Les paramètres physiques nominaux considérés sont une demie longueur de tige L de 0.26m, une masse de la tige m de 0.37Kg et une masse du chariot M de 1.4Kg.

Un module optionnel est utilisable sur l'unité actionneur, il s'agit d'un simulateur de perturbations électriques permettant d'atténuer le pourcentage des signaux transmis. Les perturbations introduites sur le système réel sont donc externes (perturbation électriques sur θ , y , \dot{y} et U simulant des problèmes de dysfonctionnement des capteurs ou actionneurs ou de simples bruits interférant avec les signaux de la boucle de régulation), mais aussi internes (perturbations mécaniques sur la demie longueur ou la masse de la tige - respectivement 13.5cm, 6cm et 0.135Kg, 0.09Kg-).

1.4.2 Les fours

Plusieurs systèmes ont été considérés: des modèles linéaires et non linéaires et un simulateur non linéaire utilisé en temps réel.

A- Modèles pour les simulations

La fonction de transfert utilisée, dans un premier temps, caractérise un système linéaire du second ordre (1.9). K_s est le gain statique, T_s la constante de temps double et T_u est le retard pur du four. Cette fonction de transfert est discrétisée avec un bloqueur d'ordre zéro.

Dans un second temps, des comportements non linéaires sont simulés par introduction de perturbations internes sur la constante de temps et sur le gain des modèles, ainsi que par la modification du retard pur non modélisé.

Les perturbations sur la constante de temps consistent en des augmentations aléatoires, dans un certain domaine autour de sa valeur nominale, appliquées à intervalles de temps réguliers. La valeur du gain évolue selon la loi parabolique (1.10) en fonction de la valeur courante de la référence $-C(k)$. Donc, pour simuler cette perturbation, une suite de niveaux de consigne est appliquée dans un domaine de fonctionnement

assez grand. Cette loi a été déduite des mesures du gain sur le système de chauffage par réglette à infra rouge, pour différentes consignes, voir le tableau (7.1). Le retard pur introduit représente des erreurs de modélisation, sa valeur pouvant atteindre 40% de la valeur de la constante de temps du modèle.

Le réglage du régulateur étant effectué sur le four nominal, des tests de sensibilité et de robustesse par rapport aux perturbations considérées pourront être réalisés.

$$G(p) = \frac{K_s * e^{-T_d p}}{(1 + T_s p)^2} \quad (1.9)$$

$C(k)$	75.5°	90°	100°	105°	115°
K_s	3.3	1.7	5.3	2.2	4.8

$$K_s = 0.00035 * C(k)^2 \quad (1.10)$$

Tableau 1.1 : Variations de K_s .

B- Dispositif expérimental de la réglette à infra rouge

Le simulateur non linéaire considéré est constitué d'une réglette à infra rouge reposant sur des briques réfractaires, voir la figure (1.9). Un capteur de température relié au PC, via le régulateur industriel *microcore III* de la société CORECI, permet d'acquérir les mesures au niveau de la réglette. Un étage de puissance transmet à la réglette la tension de commande, en provenance du PC via le *microcore*. Celui-ci sert d'interface de communication et d'interface d'affichage des mesures et commandes.

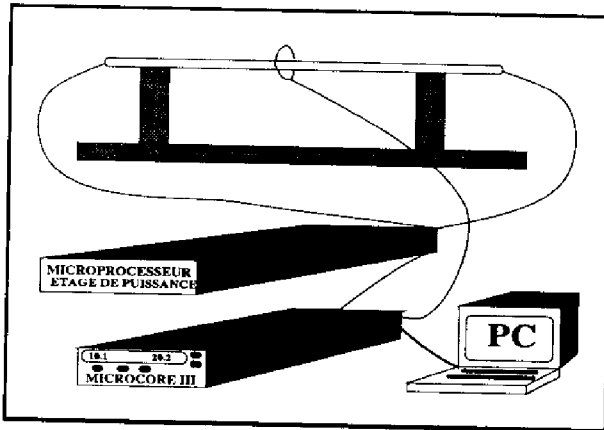


Figure 1.9 : Four à réglette à infra rouge.

Le modèle de ce système n'a pas été complètement identifié, car ceci n'est pas nécessaire pour la conception de la loi de régulation. Toutefois des éléments sur les non-linéarités propres à ce dispositif sont connues: le système possède un retard et son gain est variable, comme l'indiquent les mesures effectuées en fonction du niveau de consigne sur le tableau (7.1).

Nous avons présenté ici les problèmes généraux de l'Automatisation des procédés nous amenant à nous intéresser aux techniques floues et neuronales. Nous avons également décrit les systèmes expérimentaux qui ont été utilisés dans cette thèse. Nous allons maintenant détailler les principes de la régulation floue.

Chapitre 2

Régulation floue: Méthodologies

Ce chapitre a pour but de présenter les approches de conception d'un RF de la première génération. Les notions de base nécessaires à la compréhension de la régulation floue sont rappelées. Nous synthétisons les choix possibles pour les nombreux paramètres du régulateur flou. Nous proposons également une technique pour une mise en oeuvre efficace d'un régulateur flou sous forme d'une table de décision.

Le paragraphe (2.1) introduit les domaines scientifiques autour de la régulation floue et la structure des régulateurs flous. Le paragraphe (2.2) présente les choix possibles pour les paramètres des RF et leur ordre de réglage. Finalement, le paragraphe (2.3) aborde des problèmes théoriques (stabilité et robustesse) et pratique (inférence hors ligne ou en ligne) de mise en oeuvre des régulations floues.

2.1 Introduction aux techniques floues

2.1.1 Domaines scientifiques autour de la régulation floue

Le rôle du présent paragraphe est de situer le domaine de la régulation floue dans un contexte général, en introduisant les domaines scientifiques principaux reliés aux RF et certaines notions de base.

Il faut noter que la commande floue n'est pas l'application principale des théories du flou. Plusieurs autres secteurs d'applications peuvent être énumérés, dont les *bases de données* (avec des requêtes flexibles), les problèmes *d'ordonnancement*, *l'optimisation sous contraintes* (avec critère, contraintes flexibles et agrégation de plusieurs critères), la *classification*, les méthodes de *calcul* avec des données incomplètes, les problèmes de *prise de décision* dans des contextes imprécis et incertains, etc.... [BOU,93] et [DUB,91] permettent de compléter les informations données ci-dessous.

A- Contexte général

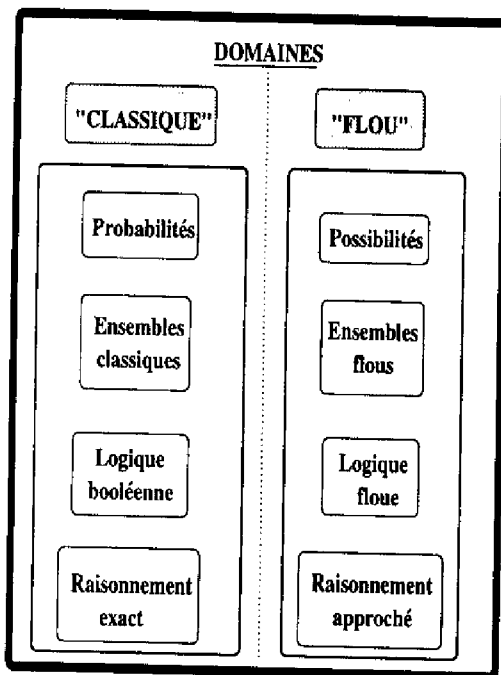


Figure 2.1 : Domaines théoriques liés au flou.

Les quatre théories principales du flou, permettant de traiter les problèmes utilisant des connaissances incomplètes, sont définies dans la suite, voir la figure (2.1). Premièrement, la *théorie des possibilités* définit les concepts de base pour traiter les connaissances incomplètes. Deuxièmement, la *théorie des ensembles flous* permet de coder les valeurs de variables linguistiques sous forme de distributions de possibilité sur des référentiels appropriés (appelés *univers de discours*). Troisièmement, la *logique floue* offre des outils de manipulation logique d'informations, à partir de la représentation sémantique proposée par la théorie des EF. Enfin, le *raisonnement approché* permet d'opérer des déductions avec pour support les opérateurs de la logique floue.

Les dates importantes relatives au développement des techniques floues sont:

années 50: *Shackle* développe un modèle possibiliste pour l'économie;

1965: *Zadeh* introduit la théorie des sous-ensembles flous;

1972: *Zadeh* et *Chang* posent les principes de base de la régulation floue;

1973: *Kaufmann* présente la notion d'EF en France;

1975: *Mamdani* développe le premier régulateur flou sur un générateur de vapeur, [MAM,75];

fin des années 70:

- *Zadeh* introduit la théorie des possibilités, puis *Dubois & Prade* la développent,
- *Zadeh* introduit la Logique floue,
- beaucoup d'applications de la RF en Europe, puis baisse d'intérêt;

à partir de 1980: développement des études sur les EF en Europe;

1985: *Togai & Watanabe* construisent le premier circuit VLSI flou aux Etats Unis, les processeurs flous sont en pleine expansion (*Yamakawa, Omron, Fujitsu, Hitachi...*);

à partir de 1985: grand développement des applications des régulateurs flous au Japon;

à partir de 1991: regain d'intérêt pour les techniques de régulation floue en Europe, opérations médiatiques, domaine reconnu par les communautés scientifiques (ex.: création du groupe *Commande symbolique* du *GDR Automatique...*).

B- Quelques éléments sur la théorie des sous-ensembles flous

Le mode de représentation utilisant les EF est le *formalisme linguistique*. Le formalisme linguistique est un langage défini pour manipuler des mots, aussi appelés *variables linguistiques* ou *variables floues*. Les variables floues caractérisent des attributs de certains objets ou personnes. Un exemple peut être l'objet *voiture* et l'attribut *vitesse*. Les valeurs possibles des variables floues, dites *valeurs floues*, sont exprimés par des *labels d'EF*, comme par exemple, {*grande, moyenne, petite*}. Les définitions numériques des EF sont faites sur les *univers de discours*, représentant les échelles de valeurs de l'attribut considéré. Deux types de règles sont donc nécessaires pour fixer le formalisme linguistique: une règle syntaxique permettant de générer les labels des ensembles flous, décrivant les valeurs d'une variable linguistique donnée, et une règle sémantique associant à chaque valeur floue un sens numérique, [KAU,75] [ZAD,65].

La notion fondamentale d'*ensemble flou* est une extension du concept d'ensemble. Par exemple, l'attribut *jeune* peut être représenté par la fonction caractéristique d'un ensemble défini sur l'intervalle $[0, 25]$ ans, voir la figure(2.2). L'introduction nécessaire de gradualité, dans la traduction mathématique de *jeune*, est envisageable avec l'utilisation de la fonction d'appartenance d'un ensemble flou. Cette fonction mathématique exprime les degrés de vérité de l'appartenance des *âges* au concept *jeune*, [BOU,93]. Avec cette représentation, l'appartenance à un ensemble peut être partielle; le lien entre une variable x et une valeur floue F est d'autant plus fort que $\mu_F(x)$ est élevé. Les EF permettent de modéliser des catégories non définies précisément et dont la signification numérique dépend du contexte.

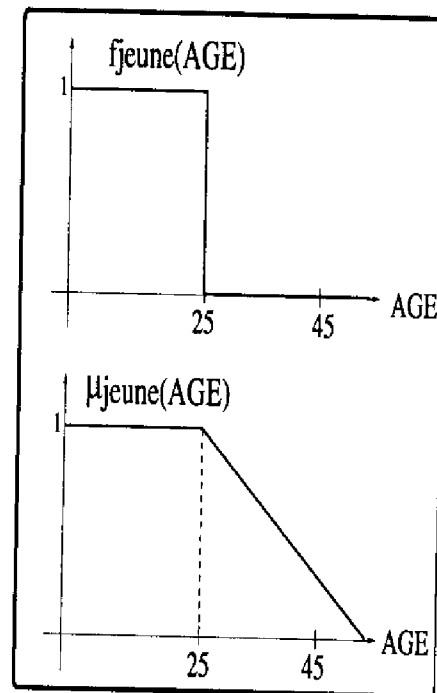


Figure 2.2 : Fonction caractéristique d'un ensemble classique et fonction d'appartenance d'un ensemble flou.

Un ensemble flou F est composé de couples $\{(x), \mu_F(x)\}$. x est l'élément générique appartenant à un univers de discours X pouvant être discret ou continu. $\mu_F(x)$ est le degré d'appartenance de cet élément à l'ensemble flou F ; $\mu_F(x)$ est compris entre 0 et 1. Certaines caractéristiques d'un EF sont -voir la figure (2.3)-: sa hauteur H (valeur d'appartenance la plus importante sur X , caractérisant les données les plus significatives), son support S (ensemble des points de X qui ont une appartenance non nulle à F), son noyau N (ensemble des points de X qui ont une appartenance totale à F).

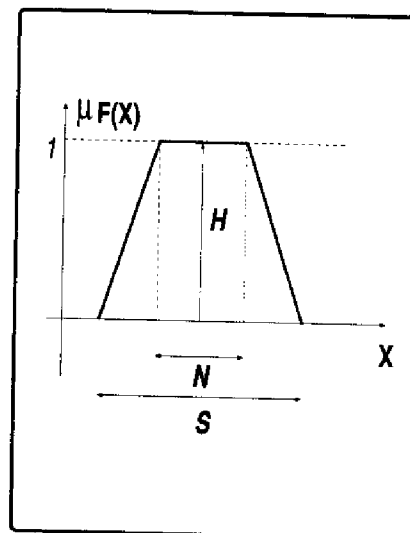


Figure 2.3 : Caractéristiques des EF.

Un ensemble flou très spécial est le *singleton flou*. Il est tel que son support est

constitué par un seul point de X qui a un degré d'appartenance égal à 1, voir l'équation (2.1).

$$\left\{ \begin{array}{l} \text{SI } x = x_0 : \mu_F(x_0) = 1 \\ \text{SINON} : \mu_F(x) = 0 \end{array} \right. \quad (2.1)$$

C- Quelques éléments sur la théorie des possibilités

Le modèle des possibilités est complémentaire de celui des probabilités, qui possède certaines limites. En effet, les outils statistiques nécessitent des hypothèses simplificatrices qui ne sont pas toujours réalistes. De plus, ils ne permettent pas de décrire toutes les situations, notamment *l'ignorance partielle ou totale*, ou les cas où *l'information n'est pas assez complète*.

L'information incomplète est soit incomplètement spécifiée (imprécise ou vague), soit non fiable (incertaine).

Une information est imprécise s'il est impossible de répondre précisément à une question: par exemple, *Quelle est la température?* (réponse possible: *entre 20° et 30° C*), [BOU,93]. Elle traduit ce qui est mal connu, c'est-à-dire les valeurs possibles/impossibles, exprimables sous forme de degrés de vérité. La notion d'imprécision dépend du contexte, du grain du référentiel. Une information vague possède des frontières mal définies (ex.: *il fait environ 18° C*).

Une information est incertaine s'il est impossible de répondre à la question: *est-ce vrai ou faux?*, par exemple: *viendra-t-il aujourd'hui?*, [BOU,93]. L'incertitude traduit un doute sur la validité d'une information.

Précision et incertitude sont des concepts opposés, une information est difficilement précise et certaine à la fois (ex.: *il arrivera à 10h22'33"*); la précision peut se définir à un certain degré de certitude.

La théorie des possibilités est un outil mathématique supplémentaire pour aider à décrire le monde réel. Le choix d'un outil dépend donc de la nature de l'information à manipuler. Les probabilités sont adaptées à des informations précises à caractère fréquentiel, obtenues d'après des expériences. Les possibilités permettent de coder plus facilement des degrés de possibilité d'événements donnés par des experts et délimitant des probabilités inconnues, [BOU,93].

Les mesures de la théorie des possibilités, la *Nécessité* et la *Possibilité*, sont en effet des bornes inférieure et supérieure des probabilités, lorsque celles-ci existent: une éventualité doit être possible avant d'être probable et probable avant d'être certaine. Une mesure de possibilité caractérise la faisabilité d'un événement, le degré auquel un fait est compatible avec la connaissance disponible. La nécessité traduit le niveau de certitude en exprimant le degré d'impossibilité de l'événement contraire. L'ignorance totale est caractérisée par une mesure de possibilité égale à l'unité et donc par une mesure de nécessité égale à zéro.

D- Quelques éléments sur la Logique floue

La logique floue permet de manipuler des degrés de vérité appartenant à l'intervalle $[0, 1]$. De nouveaux opérateurs doivent être définis pour manipuler ces nouvelles notions, [LEE,90]. Le morphisme liant la théorie des ensembles classiques à la logique propositionnelle par les fonctions caractéristiques est étendu au flou, voir les équations (2.2), où A , B , F_1 et F_2 sont respectivement deux ensembles classiques et deux EF, d'éléments de base x . Les nouveaux opérateurs *T-norme* (norme triangulaire) ou *conjonction floue* et *S-norme* ou *T-conorme* (conorme triangulaire) ou *disjonction floue*, représentent l'intersection et l'union d'EF. La négation floue est également décrite en (2.2). Les équations (2.3) et (2.4) présentent respectivement les opérateurs T-norme et T-conorme; a et b appartiennent à $[0, 1]$.

$$\left\{ \begin{array}{l} f_{A \cap B}(x) = f_A(x) \wedge f_B(x) \\ f_{A \cup B}(x) = f_A(x) \vee f_B(x) \end{array} \right. \quad \left\{ \begin{array}{l} \mu_{F_1 \cap F_2}(x) = T[\mu_{F_1}(x), \mu_{F_2}(x)] \\ \mu_{F_1 \cup F_2}(x) = S[\mu_{F_1}(x), \mu_{F_2}(x)] \end{array} \right.$$

$$\mu_{\overline{F_1}}(x) = 1 - \mu_{F_1}(x) \quad (2.2)$$

$$\left\{ \begin{array}{ll} \text{minimum:} & a \wedge b = \text{MIN}[a, b] \\ \text{produit algébrique:} & a \cdot b = ab \\ \text{produit borné:} & a \odot b = \text{MAX}[0, a + b - 1] \\ \text{produit drastique:} & a \sqcap b = \begin{cases} a & \text{si } b = 1 \\ b & \text{si } a = 1 \\ 0 & \text{si } a < 1 \text{ et } b < 1 \end{cases} \end{array} \right. \quad (2.3)$$

$$\left\{ \begin{array}{ll} \text{maximum:} & a \vee b = \text{MAX}[a, b] \\ \text{somme algébrique:} & a \hat{+} b = a + b - ab \\ \text{somme bornée:} & a \oplus b = \text{MIN}[1, a + b] \\ \text{somme disjointe:} & a \Delta b = \text{MAX}\{\text{MIN}[a, 1 - b], \text{MIN}[1 - a, b]\} \\ \text{somme drastique:} & a \sqcup b = \begin{cases} a & \text{si } b = 0 \\ b & \text{si } a = 0 \\ 1 & \text{si } a > 0 \text{ et } b > 0 \end{cases} \end{array} \right. \quad (2.4)$$

Une dualité existe entre les deux types d'opérateurs; ils sont liés par la loi de *Morgan* - $S(a, b) = 1 - T(1 - a, 1 - b)$ -. Toutefois cette relation n'oblige pas à les utiliser selon des combinaisons pré-définies. Toutes les propriétés de l'algèbre de *Boole* ne sont pas conservées avec les opérations floues; les choix des combinaisons d'opérateurs dépendent des propriétés désirées. Le choix le plus proche de la logique booléenne est le *minimum* comme T-norme et le *maximum* comme S-norme.

L'implication floue est un type particulier de relation floue. Une relation floue exprime un lien entre les éléments d'univers de discours différents, elle est caractérisée par la donnée de couples ordonnés (x, y) et des degrés d'appartenance correspondants $\mu_R(x, y)$; une représentation matricielle peut synthétiser ces données.

Une implication floue (x est $X_i \rightarrow y$ est Y_j) est caractérisée par un degré de vérité, correspondant au poids de la relation entre l'antécédent et la conclusion de la règle. Plusieurs fonctions d'implication floue ont été définies dans la littérature, par généralisation de l'implication de la logique classique, [BOU,93]. Trois grandes classes d'implications floues pures peuvent être distinguées.

La première est l'extension $S[1 - \mu_{X_i}(x), \mu_{Y_j}(y)]$ de l'égalité de la logique classique: $X \rightarrow Y = \bar{X} \vee Y$. Un exemple est l'implication de *Dienes*: $MAX[1 - \mu_{X_i}(x), \mu_{Y_j}(y)]$. La seconde classe est basée sur la définition: $T^*[\mu_{X_i}(x), \mu_{Y_j}(y)]$ (où T^* est la quasi inverse de la T-norme T), [BOU,93]. L'idée de base est qu'il est impossible de déduire quelque chose qui est moins vrai que l'antécédent. Une illustration est l'implication de *Godel*: 1 si $\mu_{X_i}(x) \preceq \mu_{Y_j}(y)$, sinon $\mu_{Y_j}(y)$. La troisième classe étend l'égalité de la logique classique: $X \rightarrow Y = \bar{X} \vee (X \wedge Y)$. L'implication de *Dienes* peut résulter également de cette définition.

Des relations floues de type implication floue ne sont pas toujours traduites par des dérivées de l'implication booléenne (particulièrement en commande). Ainsi, la conjonction de *Mamdani*, utilisant l'opérateur *minimum*, est souvent appelée abusivement *implication floue*.

E- Quelques éléments sur le raisonnement approché

a- Présentation générale: Deux formes principales de raisonnement peuvent être définies. La première est un mécanisme de raisonnement par chaînage avant: de la connaissance, non explicitement connue, est déduite. Le raisonnement est guidé par les données, en ce sens que les entrées permettent d'atteindre les conclusions de certaines implications. La règle d'inférence qui permet de réaliser ce raisonnement est le *Modus Ponens*. La deuxième forme de raisonnement s'effectue par chaînage arrière. La règle d'inférence est le *Modus Tollens*, qui permet de remonter aux causes de certaines manifestations. Dans les régulateurs flous, seul le mécanisme de chaînage avant à un pas est utilisé.

Avec le *Modus Ponens* de la logique classique, le fait proposé en prémisses d'un raisonnement doit vérifier exactement la condition d'une implication, pour que la conclusion de l'implication considérée puisse être déduite.

Une extension a été faite au cas flou par *Zadeh* afin de pouvoir calculer l'image d'un EF par une relation floue: c'est le *Modus Ponens Généralisé*. Cette forme de raisonnement est présenté dans la relation (2.5).

$$\begin{array}{l}
 \text{Implication: SI } x \text{ est } X_i \text{ ALORS } y \text{ est } Y_j \\
 \text{Fait: } x \text{ est } X_i^* \\
 \hline
 \text{Conclusion: } y \text{ est } Y_j^*
 \end{array} \quad (2.5)$$

Le problème de l'inférence floue consiste à composer les connaissances de la valeur floue en entrée X_i^* avec les valeurs de vérité de la relation floue considérée, de façon à établir les possibilités pour que y prenne certaines valeurs. Il s'agit de trouver Y_j^* cohérent avec la connaissance disponible et avec des critères logiques approchés du tableau (2.1), tels que plus x est différent de A , moins y est défini précisément. Un fait différent de la conclusion d'une implication peut donc être conclu, lorsque le fait proposé en prémisse est proche de la condition exprimée dans l'antécédent de la règle considérée. Il faut calculer le degré d'adéquation entre l'implication et le fait donné.

Fait: x est	A	beaucoup A	plus ou moins A	non A
Conclusion: y est	B	beaucoup B	plus ou moins B	inconnu non B

Tableau 2.1 : Critères logiques du raisonnement approché.

b- Déroulement mathématique du raisonnement approché: Il s'agit de caractériser la fonction d'appartenance $\mu_{Y_j^*}(y)$. Le principe de base est que le degré de vérité de " y est Y_j^* " est *au moins aussi grand* que le degré de " x est X_i^* " et que celui de " $X_i \rightarrow Y_j$ ". Il en résulte la règle compositionnelle d'inférence *Sup-T*, où T est une norme triangulaire, voir l'équation (2.6). Cette règle est une composition entre une relation floue R_{ij} définie sur $(X \times Y)$ et une variable linguistique x définie sur X . La règle d'inférence peut également être justifiée à partir du principe de combinaison-projection (permettant de calculer l'image d'un intervalle par un graphe classique), [DUB,91].

$$\begin{cases} Y_j^* & = X_i^* \circ R_{ij_{X_i \rightarrow Y_j}} \\ \mu_{Y_j^*}(y) & = \sup_{x \in X} T[\mu_{X_i^*}(x), R_{ij_{X_i \rightarrow Y_j}}] \end{cases} \quad (2.6)$$

L'utilisation des quatre opérateurs de type T-norme conduit à la définition des quatre règles compositionnelles d'inférence: *sup_min*, *sup_.*, *sup_⊙* et *sup_⊓*. Pour des raisons de simplification des calculs, les deux premières sont les plus souvent utilisées. Toutefois, les deux dernières permettent d'obtenir des résultats inférés plus précis pour le raisonnement approché, [MIZ,88].

c- Les étapes de l'inférence: L'équation (2.6) montre que la première étape du processus d'inférence est la détermination de la matrice, caractérisant l'implication floue R_{ij} . Cette matrice a pour indices d'entrée les colonnes x et les lignes y et pour contenu $R_{ij} x_i \rightarrow y_j(x, y)$. La deuxième étape de l'inférence consiste à composer les connaissances de la valeur floue d'entrée de x et de l'implication floue. Une conjonction est opérée entre le contenu de la matrice précédente et l'EF $(x, \mu_{x_i}(x))$, pour chaque valeur de l'index y . Enfin, l'étape finale revient à générer l'EF $(y, \mu_{y_j}(y))$, à partir des valeurs maximales, sur chaque ligne, de la matrice obtenue à l'étape précédente.

Plus généralement, la connaissance peut être inférée à partir de plusieurs relations d'implication possédant plus d'un antécédent; les équations (2.7) et (2.8) illustrent ce cas de figure. Deux autres étapes s'ajoutent alors au processus d'inférence. La première consiste à donner un sens sémantique au *et*, caractérisant la connaissance commune aux deux variables de la partie antécédent d'une implication. La dernière étape consiste à agréger les résultats inférés pour les différentes règles tirées. En fait, dans le cas général, les faits en entrée sont combinés à l'ensemble de la base de règles. Toutefois, avec les règles d'inférence *sup*- et une agrégation de type *maximum*, les opérateurs d'inférence et d'agrégation sont commutatifs, [LEE,90]. Ce cas de figure correspond à l'équation (2.8).

$$\begin{array}{l}
 \text{Implication } R_{i,j,k}: \text{ SI } e \text{ est } E_i \quad \text{ET } \Delta e \text{ est } \Delta E_j \quad \text{ALORS } U \text{ est } U_k \\
 \text{Faits:} \quad \quad \quad e \text{ est } E^* \quad \quad \text{ET } \Delta e \text{ est } \Delta E^* \\
 \hline
 \text{Conclusion:} \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad U \text{ est } U_k^*
 \end{array}$$

$$\begin{array}{l}
 R_{i-1,j-1,k-1}: \text{ SI } e \text{ est } E_{i-1} \quad \text{ET } \Delta e \text{ est } \Delta E_{j-1} \quad \text{ALORS } U \text{ est } U_{k-1} \\
 \text{Faits:} \quad \quad \quad e \text{ est } E^* \quad \quad \text{ET } \Delta e \text{ est } \Delta E^* \\
 \hline
 \text{C:} \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad U \text{ est } U_{k-1}^*
 \end{array} \quad (2.7)$$

$$\left\{ \begin{array}{l}
 \mu_{E_i \Delta E_j}(e, \Delta e) = ET[\mu_{E_i}(e) , \mu_{\Delta E_j}(\Delta e)] \\
 R_{\{E \text{ et } \Delta E \rightarrow U\}}^{i,j,k}(e, \Delta e, u) = Impli[\mu_{E_i \Delta E_j}(e, \Delta e) , \mu_{U_k}(u)] \\
 \mu_{U_k^*}(u) = \sup_{e, \Delta e} T[\mu_{E_i^*}(e) , \mu_{\Delta E_j^*}(\Delta e) , R_{\{E \text{ et } \Delta E \rightarrow U\}}^{i,j,k}(e, \Delta e, u)] \\
 \mu_{U^*}(u) = AGREG_{k=1}^n [\mu_{U_k^*}(u)]
 \end{array} \right. \quad (2.8)$$

d- Simplification de l'inférence lors de l'utilisation de singletons: Pour des entrées *singletons*, le mécanisme d'inférence est considérablement simplifié.

$$\left\{ \begin{array}{l}
 \mu_{U^*}(u) = MAX_{k=1}^n [MIN(\alpha_k, \mu_{U_k}(u))] \\
 \alpha_k = MIN[\mu_{E_i}(e_o), \mu_{\Delta E_j}(\Delta e_o)]
 \end{array} \right. \quad (2.9)$$

Par exemple, avec la conjonction floue de *Mamdani*, le *et* traduit par le minimum, l'agrégation exprimée par le maximum et la règle compositionnelle d'inférence de *Zadeh*, les fonctions d'appartenance du résultat inféré peuvent s'exprimer simplement. Les entrées sont e_0 et Δe_0 ($\mu_E(e_0) = 1$ et $\mu_{\Delta E}(\Delta e_0) = 1$). L'équation (2.9) exprime le résultat inféré. E_i et ΔE_j sont les ensembles flous proposés dans la partie antécédent de la règle R_{ij} , α_k est le poids exprimant le degré de contribution de la règle R_{ij} à l'action de commande floue inférée, U_k est la valeur floue dans la conclusion de la règle R_{ij} , et n est le nombre de règles tirées.

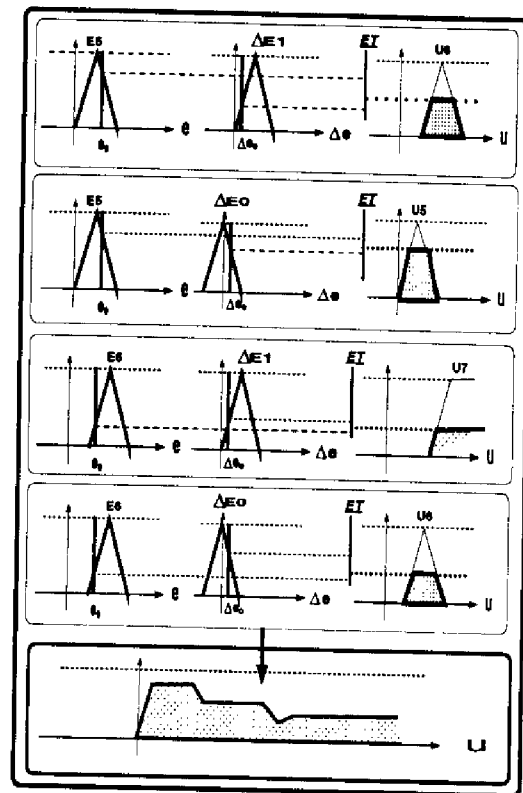


Figure 2.4 : Inférence avec singletons.

2.1.2 Présentation générale des régulateurs flous

A- Description générale des régulateurs flous

La relation entrée-sortie d'un régulateur flou est numérique, même si elle est établie à partir du traitement de relations linguistiques faisant intervenir des EF. La loi de régulation numérique obtenue peut être de type PD, PI ou PID, suivant les entrées et sorties considérées. La figure (2.5) présente différentes manières de réaliser l'une de ces trois formes de relation entrée-sortie.

Le schéma (2.6) décrit les différents modules du régulateur flou. Dans les paragraphes suivants sont décrits plus précisément les modules du RF.

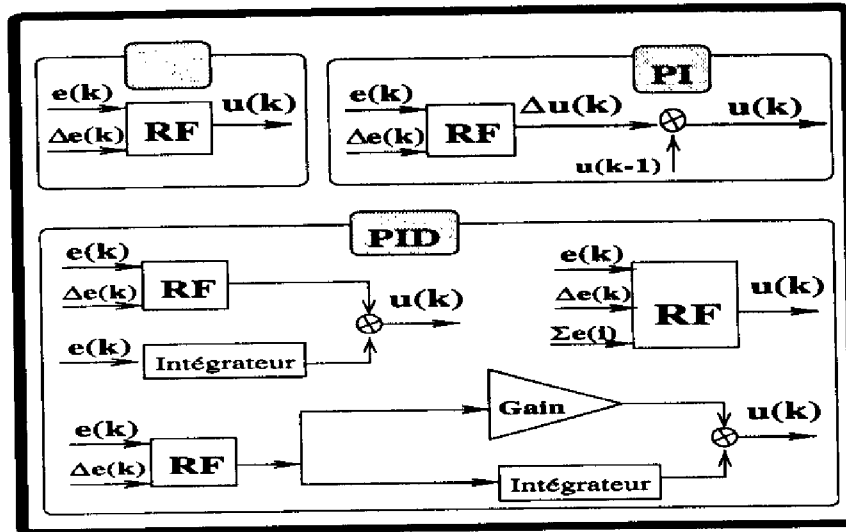


Figure 2.5 : Réalisations de lois de régulation floue de type PD, PI ou PID.

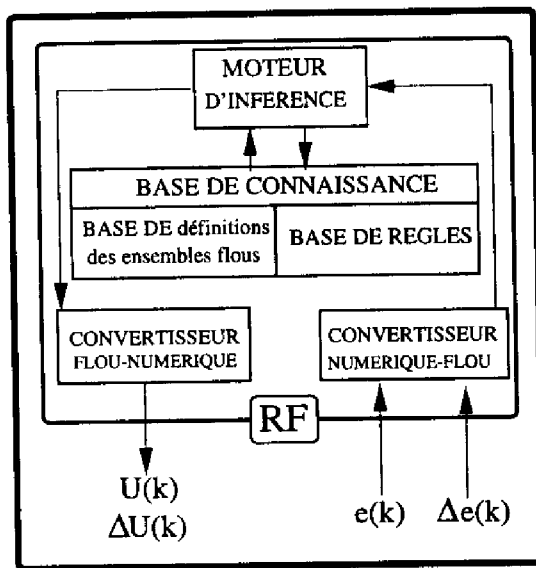


Figure 2.6 : Schéma global d'un RF.

Les entrées numériques du régulateur flou, caractérisant l'évolution des sorties du système, transitent d'abord par le *convertisseur Numérique-Flou*. Converties en valeurs floues, elles constituent les faits en entrée du *moteur d'inférence*. Ensuite, ce dernier applique la règle compositionnelle d'inférence en utilisant la *base de règles* et la *base de définitions des ensembles flous*. La valeur floue déduite pour la sortie du régulateur $-U(k)$ ou $\Delta U(k)$ - est convertie en une commande numérique par le *convertisseur Flou-Numérique*. L'ensemble du cycle de calcul du régulateur peut recommencer à partir de nouvelles mesures.

B- Convertisseurs Numérique-Flou et Flou-Numérique

Les interfaces Numérique-Flou et Flou-Numérique gèrent les transformations entre données numériques et quantités floues et inversement.

La première interface s'occupe de l'aspect *fuzzification* des variables d'entrée. Deux types de fuzzification peuvent être décrites, [FOU,92]. La première est une *fuzzification*

numérique; elle associe un ensemble flou à une entrée numérique. Cette forme de fuzzification est utilisée pour des mesures numériques faites par des capteurs imprécis ou précis (cas du codage avec des singletons). La deuxième forme de *fuzzification* est *symbolique*; elle consiste à définir l'*interprétation* d'une mesure numérique, c'est-à-dire une description symbolique de la mesure avec des labels d'EF. Cette opération est réalisée notamment par les capteurs flous. L'opération de fuzzification symbolique correspond, en fait, à une étape de filtrage flou conditions-faits (*pattern matching*), entre un singleton en entrée et les ensembles flous présents dans les antécédents des règles.

La seconde interface réalise une *défuzzification*. Cette opération a pour objectif de transformer une décision floue en une valeur numérique acceptable par les actionneurs. De même que précédemment, il existe deux approches, [FOU,92]. La *défuzzification numérique* fait correspondre à un EF défini sur un univers de discours X , une des valeurs de X . Il s'agit de transformer une donnée floue en une donnée numérique traduisant au mieux la distribution de possibilités inférée pour la sortie du régulateur. La *défuzzification symbolique* fait correspondre à un ensemble d'interprétations symboliques de données, un EF, auquel il est possible d'appliquer ensuite une défuzzification numérique.

C- Base de connaissance et moteur d'inférence

La base de connaissance est composée de la base de définitions des ensembles flous et de la base de règles. L'expression naturelle des connaissances nécessite les étapes de définition des variables linguistiques, de l'ensemble de leurs valeurs floues, de leur univers de discours et des règles floues.

La base de définitions des variables floues permet d'établir une correspondance entre valeurs floues et sens numérique. Elle contient les définitions des EF. Dans ce module sont précisés les *bornes des univers de discours* pour les variables E-S du régulateur flou. Les manipulations mathématiques étant opérées dans des intervalles normalisés, les définitions des domaines de variation des signaux normalisés et réels sont nécessaires. Ces informations sont parfois données par des *facteurs d'échelle* entre les données réelles en entrée et les valeurs sur les univers de discours normalisés. Les valeurs floues des variables E-S sont également définies dans cette base de définitions, ainsi que le nombre d'EF pour chaque variable floue, leur forme et leur répartition sur leur univers de discours respectifs.

La construction de la base de règles pose des problèmes syntaxiques et sémantiques, concernant respectivement les choix du type de règles floues et de leur contenu. La connaissance linguistique est exprimée dans des règles floues de la forme (2.10), où E_i , ΔE_j et U_k sont des labels d'EF, e et Δe sont les variables d'entrée, et u est la variable de sortie du RF. La partie antécédent est une description de l'état du système; la partie conséquence présente les actions à exercer sur le système à réguler. La relation floue correspondante n'est autre qu'une implication floue, dont il a été question dans le paragraphe (2.1.1), voir la formule (2.11).

$$R^{i,j,k} : \text{SI } [(e \text{ est } E_i) \text{ ET } (\Delta e \text{ est } \Delta E_j)] \text{ ALORS } (u \text{ est } U_k) \quad (2.10)$$

$$R^{i,j,k} : \mu_{\{(E_i \text{ et } \Delta E_j) \rightarrow U_k\}}(e, \Delta e, u) = \text{Impli}\{ \text{ET}[\mu_{E_i}(e), \mu_{\Delta E_j}(\Delta e)], \mu_{U_k}(u) \} \quad (2.11)$$

Les différentes règles de la base sont agrégées. Un système flou est donc caractérisé par une combinaison de relations floues, selon (2.12).

$$\text{BASE} = \text{AGREG}_{i,j} [R^{i,j,k}] \quad (2.12)$$

Dans le cas particulier des règles de type *Sugeno*, les conclusions sont une fonction analytique des variables d'entrée. Mis à part les règles à évaluation d'état, permettant de dériver, à un instant donné, une action à partir des mesures faites à l'instant précédent, il existe les règles à évaluation d'objet, contenues dans les régulateurs prédictifs flous. Elles permettent de prédire le résultat des actions de commande en évaluant des indices de performances traduisant des objectifs de régulation.

Le moteur d'inférence permet de réaliser les étapes logiques du raisonnement approché. Il nécessite le choix des opérateurs mathématiques interprétant les fonctions logiques citées ci-après: le **et** exprimant le lien entre les variables linguistique des antécédents des règles, la **fonction d'implication floue** reliant les variables d'entrée et de sortie du système flou, la **règle compositionnelle d'inférence** combinant les informations floues, et l'agrégation réunissant les différentes règles.

2.2 Conception des régulateurs flous

Après avoir décrit les modules composants les RF de manière générale, l'objet de cette partie est de décrire différentes stratégies de conception et de mettre en évidence l'importance relative des différents paramètres, lors de l'étape de réglage des régulateurs, voir [LEE,90] pour une description exhaustive.

2.2.1 Paramètres des convertisseurs N-F et F-N

A- Convertisseurs N-F

Concernant les méthodes de fuzzification numérique, deux stratégies sont principalement utilisées.

- Une des méthodes consiste à générer des données probabilistes en utilisant un bruit aléatoire sur les signaux à mesurer. Ensuite, les dispersions des mesures sont calculées avec des méthodes statistiques. Les données probabilistes obtenues sont enfin converties en données possibilistes, [LEE,90].
- Une méthode de fuzzification plus couramment employée, pour des raisons de facilité de mise en oeuvre, consiste à transformer les valeurs discrètes en singletons flous sur l'univers de discours de la variable d'entrée en cours de traitement. Dans ce cas, le terme *fuzzification* est parfois attribué à la double fonction de correspondance entre données discrètes numériques et singletons *et* d'intersection entre ces singletons et les EF présents dans les antécédents des règles.

B- Convertisseurs F-N

Il existe quatre stratégies principales de défuzzification.

- La méthode du *premier maximum* fournit la valeur où la fonction d'appartenance atteint pour la première fois sa valeur maximale lorsque l'univers de discours est parcouru dans le sens croissant.
- Dans la méthode de la *moyenne des maxima*, la valeur choisie correspond à la valeur moyenne des actions qui conduisent à une fonction d'appartenance maximale.
- La méthode du *centre de gravité* calcule le barycentre de la distribution de possibilité générée.
- Il est possible d'appliquer une *moyenne pondérée* des centres des EF inférées, dans le cas d'ensembles flous convexes et symétriques autour du noyau. Cette approche revient à défuzzifier séparément chaque règle avant de combiner les résultats de toutes les règles inférées, voir l'équation (2.13). u_i est le centre de gravité de l'ensemble flou inféré d'indice i (il correspond à la valeur médiane du support de l'ensemble flou), $\mu(u_i)$ est le degré d'appartenance de u_i à l'ensemble flou d'indice i . Cette technique permet d'obtenir des résultats proches de ceux obtenus avec la méthode du centre de gravité, avec des calculs nettement simplifiés.

$$u(k) = \frac{\sum_i u_i \times \mu(u_i)}{\sum_i \mu(u_i)} \quad (2.13)$$

D'après *Scharf et Mandic*, la méthode du centre de gravité permet d'obtenir de meilleures performances en régime permanent alors que la méthode de la moyenne des maxima fournit de meilleurs résultats en régime transitoire où une précision moins grande est exigée, [SCH,85]. De façon générale, le calcul du centre de gravité conduit

aux résultats les plus nuancés, puisque tous les résultats inférés apportent une contribution en fonction de l'importance de leur validité. Ainsi, les discontinuités dans la commande sont évitées, ce qui est fort appréciable.

2.2.2 Paramètres de la base de règles

La phase de modélisation de la connaissance sous forme linguistique peut se faire selon deux approches principales: l'*analyse qualitative* ou la *génération automatique de règles*; ces méthodes sont décrites dans les paragraphes suivants. Une fois les règles de régulation floue obtenues, l'étape finale consiste à vérifier la cohérence (consistance et complétude) de la base de règles. La propriété de complétude est vérifiée s'il existe, pour toutes les entrées du régulateur, une règle dominante qui peut être tirée. Les règles sont consistantes si elles n'engendrent pas de contradiction dans les décisions.

A- Justification de la conception de lois de régulation floue

L'utilisation de lois de commande floue peut être justifiée en mettant en évidence les stratégies de régulation dans le plan de phase.

Considérons pour cela un problème de régulation en temps minimal, dont le but est de trouver les commandes qui minimisent le temps de convergence vers la consigne, sous certaines limitations concernant les valeurs des commandes. Pour des systèmes du second ordre, la résolution analytique du problème est aisée et conduit à deux types de trajectoires paraboliques pour chacune des valeurs de saturation de u , voir la figure (2.7). La loi de commande *tout ou rien* obtenue est optimale d'un point de vue rapidité, mais possède des limitations en robustesse.

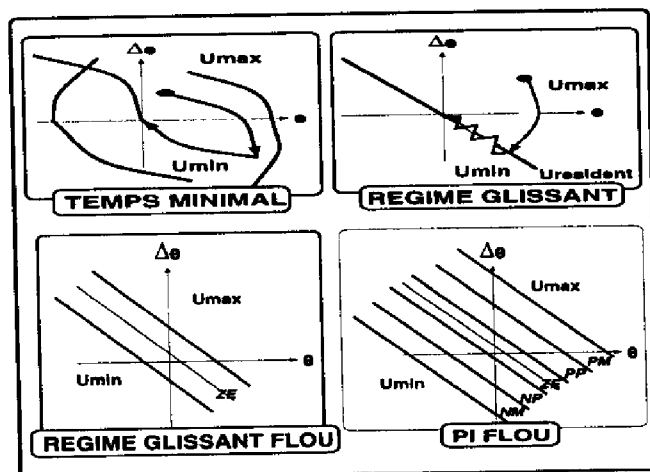


Figure 2.7 : Liens entre les lois de régulation floue, à régime glissant et PI.

Dans le cas de systèmes plus complexes, une simplification consiste à définir une droite de glissement, donc à générer une commande en *régime glissant*, moins rapide, mais plus robuste. Il est question de *commande à structure variable* pour laquelle des commutations intempestives se produisent autour de la droite de glissement, voir la figure (2.7).

Une des idées de la régulation floue est de nuancer la commande en introduisant des bandes pour réduire les commutations. En contrepartie, les vitesses de convergence vers les consignes sont réduites, les réponses ne sont plus faites en temps minimal. Le type de loi de régulation s'éloigne peu à peu de la commande en *régime glissant flou* pour converger vers la loi de régulation PI floue.

Le facteur principal qui distingue les deux cas, régime glissant flou et PI flou, est la granularité, c'est-à-dire le nombre de zones transversales et les largeurs de ces bandes. Une loi de régulation de type régime glissant est obtenue si seulement trois bandes définissent la table de règles floues, voir la figure (2.7). Les interpolations entre la diagonale *ZE* de la table et les bandes extrêmes (*NG* et *PG*), permettent de définir une courbe de glissement correspondant à la dynamique propre du système (commande nulle appliquée). Une loi de régulation de type PI saturée est obtenue si les ensembles flous des entrées et sorties sont en nombre suffisant, voir la figure (2.7).

L'introduction de non-linéarités par utilisation des nombreux degrés de liberté du RF permettra de nuancer les lois de régulation standard obtenues.

B- Analyse qualitative

Cette première méthode consiste à traduire la connaissance sous forme de règles à partir de diverses sources de connaissance, [FOU,93]. Les sources d'informations peuvent être des connaissances expertes extraites d'un manuel technique ou déduites de l'interrogation d'un expert humain, des observations des actions menées par un opérateur spécialisé sur le processus, ou des analyses de données E-S du système. Le raisonnement conduisant à l'écriture des règles peut être mené dans l'espace de phase -ou dans l'espace temporel-, mais à un niveau de description qualitatif (notion de plan de phase qualitatif ou linguistique).

Les objectifs de régulation doivent d'abord être définis. Les informations obtenues sur les dynamiques du système en boucle ouverte autorisent le choix d'objectifs de régulation, sous forme d'un modèle comportemental de référence en boucle fermée dans l'espace de travail qualitatif considéré. L'étape suivante d'écriture des règles est la définition de la stratégie de régulation permettant d'atteindre au mieux les objectifs fixés. A ces fins, pour chacune des zones qualitatives de l'espace de travail, l'évolution type nécessaire est définie; des actions qualitatives locales de commande sont alors proposées afin de converger vers la trajectoire de référence. Les figures (2.8) et (2.9) illustrent la démarche décrite précédemment; les trajectoires de référence sont représentées en courbes épaisses. Pour chaque situation (position dans le plan de phase qualitatif -caractérisé par un point- ou évolution de la réponse temporelle -caractérisé

par un point et une flèche-), est indiquée la valeur de ΔU nécessaire pour suivre au mieux la trajectoire de référence.

Une table de règles souvent utilisée est celle de *MacVicar-Whelan*, qui résulte de l'addition qualitative des termes flous en dehors des saturations ($EF_e + EF_{\Delta e} = EF_{\Delta u}$). Elle est présentée sous forme symétrique avec des bandes diagonales de niveaux de commande constants, [MAC,76]. Une analyse qualitative, à partir d'un système linéaire, permet de l'établir rapidement; elle suit la logique suivante. Si l'erreur de sortie du système et sa variation sont nulles, alors garder la commande constante. Si l'erreur de sortie est non nulle, alors l'action dépend du signe et de la valeur de e et Δe . Mais, dans le cas où l'erreur peut être corrigée naturellement alors il ne faut pas agir.

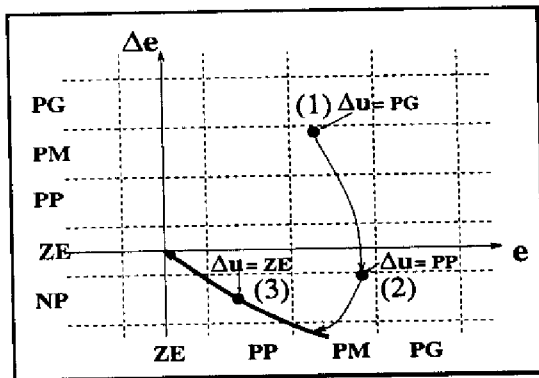


Figure 2.8 : Analyse qualitative, plan de phase.

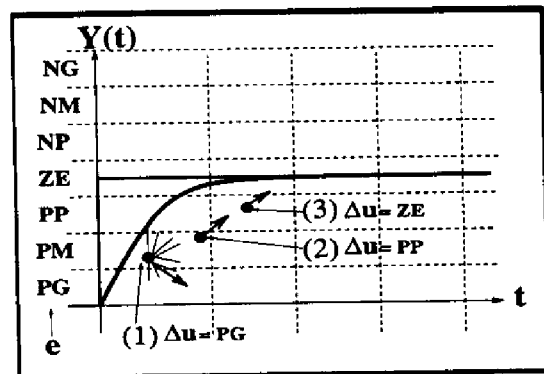


Figure 2.9 : Analyse qualitative, réponse temporelle.

C- Génération automatique

La deuxième approche pour la construction de la base de règles est fondée sur le concept d'identification floue, comprenant deux phases qui permettent une détermination systématique de la connaissance: l'identification de structure et l'estimation des paramètres des règles floues. La structure du modèle est posée a priori en définissant des conclusions de règles qui sont des fonctions des variables d'entrées (règles de Sugeno). L'estimation consiste à déterminer les paramètres de ces fonctions en résolvant un problème d'optimisation.

2.2.3 Paramètres de la base de définitions des ensembles flous

Les choix des EF et des limites des univers de discours, reposent souvent sur des approches heuristiques mais, parfois, aussi sur des règles mathématiques.

A- Définition des EF

La caractérisation complète des EF implique les choix de leurs nombres, formes et disposition sur les univers de discours.

Il est, dans un premier temps, important de choisir la *granularité floue* adéquate. A chaque variables linguistique d'entrée ou de sortie du RF est associé un ensemble de valeurs floues possibles, sous forme de labels d'ensembles flous, tels que : NG (*Négatif Grand*), NM (*Négatif Moyen*), NP (*Négatif Petit*), ZE (*Zéro*), PP (*Positif Petit*), PM (*Positif Moyen*), PG (*Positif Grand*).... Le nombre d'ensembles flous choisis détermine le nombre maximal de règles de régulation. Cette étape est réalisée d'après des connaissances intuitives sur le système. L'objectif est de minimiser le nombre de règles de régulation à concevoir, d'autant qu'un nombre élevé d'EF n'est pas nécessaire compte tenu de la nature interpolative du RF.

Après avoir défini une partition floue des domaines d'entrée et de sortie, il est nécessaire de définir la forme des différents ensembles flous, donc la définition mathématique des fonctions d'appartenance liées aux différents labels. Dans le cas où les univers de discours sont continus, une définition fonctionnelle telle une forme gaussienne, trapézoïdale ou triangulaire peut être utilisée. La forme des ensembles flous n'a qu'une très légère incidence sur les performances des régulateurs flous. La forme triangulaire permet un traitement informatique facilité du filtrage flou, tout en obtenant des résultats comparables aux résultats un peu plus nuancés liés à des formes gaussiennes. Des singletons sont parfois utilisés pour la variable de sortie du RF. Ce choix possède un certain nombre d'avantages, tout en ne dégradant pratiquement pas les résultats obtenus avec des ensembles flous plus complexes. Le premier aspect intéressant de ce choix est l'utilisation d'une méthode de défuzzification rapide de type moyenne pondérée, donc un avantage pour le calcul temps-réel. De plus, la possibilité de modifier facilement les valeurs floues de la variable de sortie est introduite par le simple déplacement des positions des singletons sur l'univers de discours.

La répartition des ensembles flous est souvent choisie uniforme. Toutefois, la disposition des EF peut ne pas être régulière; auquel cas, des procédures d'optimisation sont, en général, utilisées. La technique souvent mise en oeuvre consiste à modifier les positions des noyaux d'EF triangulaires tout en garantissant la correspondance entre le noyau d'un EF et le début du support de l'EF suivant avec $\sum_i \mu_{F_i}(x) = 1 \quad \forall x \quad \forall i$ (partition de *Bezdek*).

B- Bornes des univers de discours, indications pour un réglage heuristique

Les valeurs des bornes des univers de discours peuvent être déduites d'analyses des domaines de variation des signaux physiques, puis de réglages heuristiques en ligne, permettant alors d'affiner la définition numérique de la loi de régulation. Une autre forme de réglage, consistant à utiliser des règles d'équivalence avec les lois de régulation algorithmiques, sera décrite ultérieurement.

Les règles heuristiques utilisées pour le réglage des bornes des univers de discours d'un RF traduisent les liens perçus intuitivement entre ses paramètres et les performances dynamiques atteintes en boucle fermée. Les gains d'entrée du RF influencent principalement les performances dynamiques du système bouclé et le gain de sortie, agissant à la fois sur les deux entrées, joue un rôle prépondérant sur la stabilité.

Prenons le cas d'un régulateur flou de type PD avec la table de règles de *MacVicar-Whelan*; trois paramètres doivent être fixés, à savoir: e_{max} , Δe_{max} et u_{max} . La démarche de réglage manuel peut suivre les étapes suivantes:

1. Dans un premier temps, il est recommandé de choisir le couple $(e_{max}, \Delta e_{max})$ assurant un parcours de toute la table de règles et donc une utilisation des saturations du régulateur flou. Cette précaution permet de suivre une démarche de conception cohérente, qui ne conduise pas à l'utilisation d'une petite partie des règles développées. L'initialisation des bornes des univers de discours des variables d'entrée se fait à partir de connaissances a priori sur les domaines de variation des signaux. e_{max} peut être choisi en fonction de la condition initiale et des niveaux de consigne à atteindre. La valeur initiale de Δe_{max} peut être fixée d'après les valeurs maximales de la variation de l'erreur atteintes au cours d'une simulation-test en boucle ouverte.
2. L'étape suivante consiste à régler le niveau de saturation de la loi de commande. u_{max} est déterminé par essais successifs en fonction de la puissance maximale applicable au système et de la garantie de stabilité. Il s'agit de doser correctement l'énergie fournie par un parcours complet de la table de règles. Par exemple, une commande maximale trop faible pourra conduire à un comportement divergeant et, inversement, une valeur trop grande risquera de générer une réponse trop oscillatoire, éventuellement instable.
3. La dernière étape du réglage consiste à affiner, localement et séparément, les valeurs des gains sur l'erreur et sa variation. Le problème posé est une amélioration des performances dynamiques, temps de montée, temps de réponse..., tout en garantissant, autant que possible, un parcours complet de la table de règles. Dans le cas de la variation de l'erreur, une augmentation de sa valeur maximale tend à rendre le RF plus rapide et à diminuer les anticipations sur les dépassements: le comportement de la sortie du système devient alors plus oscillatoire. Dans ce cas, le RF est devenu proportionnellement plus sensible à l'erreur qu'à sa variation, des vitesses plus grandes de convergence de la sortie sont alors autorisées.

Un bon compromis sur les performances peut donc être obtenu expérimentalement. La démarche de réglage est facile car le RF est transparent. Par contre, la convergence vers le réglage optimal au sens d'un critère choisi sur les performances dynamiques, n'est pas garantie. Les règles d'équivalence entre paramètres du RF et des fonctions analytiques permettent alors de maîtriser davantage les opérations de réglage effectuées.

C- Bornes des univers de discours, réglage par équivalences

Les méthodes de conception décrites ici permettent de régler les paramètres de régulateurs flous, de façon à obtenir une loi de régulation équivalente à une loi analytique PI, PD ou PID. Suivant les choix des paramètres du régulateur flou, l'équivalence est plus ou moins approximative, elle peut être parfaite dans certain cas. Dans la suite, le principe général de la conception de RF par analogie à des lois de type PID est rappelé et des éléments définissant la nature plus ou moins exacte de l'approximation à une loi linéaire sont soulignés.

a- Principe d'équivalence dans le cas du PI

L'idée de base du principe d'équivalence consiste à fixer les gains équivalents - proportionnel, intégral ou dérivé- du régulateur flou par les choix des bornes des univers de discours, à partir d'un raisonnement sur les valeurs modales. Les hypothèses suivantes sont faites sur la structure du régulateur flou, [FOU,92]. Les fonctions d'appartenance des ensembles flous, caractérisant les valeurs floues des variables d'entrée et de sortie, sont de forme triangulaire et distribuées régulièrement avec un recouvrement de 1 sur les différents univers de discours ($\frac{L_e}{2} = \delta e$), voir la figure (2.10). De plus, toutes les valeurs floues des différentes variables doivent être décrites par le même nombre de labels linguistiques. La base de règles utilisée correspond à la table de *MacVicar-Whelan*, voir les formules (2.14) et (2.15) pour les notations considérées.

$$R_{i,j} : \Delta u_k = \Delta u_{i+j} = f(e_i, \Delta e_j) \quad \{i \in [1, 7]; j \in [-3, 3]; k = i + j \in [1, 7]\} \quad (2.14)$$

$$\begin{aligned} R_{4,0} &: e_4(\text{ZE}) \Delta e_0(\text{ZE}) \Delta u_4(\text{ZE}) \\ R_{4,3} &: e_4(\text{ZE}) \Delta e_3(\text{PG}) \Delta u_7(\text{PG}) \\ R_{7,3} &: e_7(\text{PG}) \Delta e_3(\text{PG}) \Delta u_7(\text{PG}) \end{aligned} \quad (2.15)$$

$$\begin{aligned} e_i^m &= (i - 4) L_e \\ \Delta e_j^m &= j L_{\Delta e} \\ \Delta u_k^m &= (k - 4) L_{\Delta u} \end{aligned} \quad (2.16)$$

$$\Delta u_{i+j} = K_p (e_i^m + K_i \Delta e_j^m) \quad \{i \in [1, 7]; j \in [-3, 3]; k = i + j \in [1, 7]\} \quad (2.17)$$

$$K_p = \frac{L_{\Delta u}}{L_e} = \frac{\Delta u_{\max}}{e_{\max}}$$

$$K_i = \frac{L_{\Delta u}}{K_p L_{\Delta e}} = \frac{\Delta u_{\max}}{K_p \Delta e_{\max}} \quad (2.18)$$

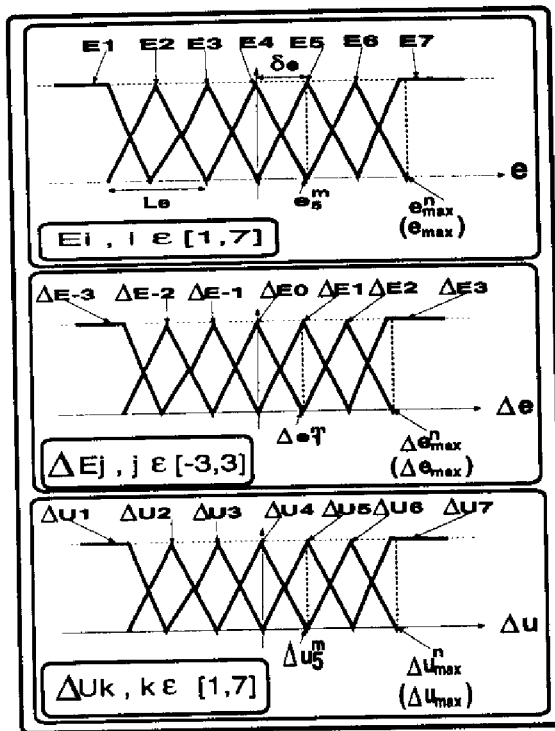


Figure 2.10 : Notations utilisées.

Pour une règle donnée, les valeurs modales correspondantes aux EF des parties antécédent et conclusion sont définies par les relations (2.16). Quand les entrées correspondent à ces valeurs, la possibilité de s'identifier avec les conditions d'une unique règle est maximale, par conséquent, la sortie est exactement égale à la conclusion. De même, pour les points de recouvrement 0.5, les interpolations résultantes du traitement des quatre règles activées se compensent, conduisant à une situation identique au cas précédent. Pour tous ces points (*points caractéristiques*), la relation E-S est exactement linéaire en dehors des zones de saturation, voir la formule (2.17). La prise en compte des relations (2.16) dans cette dernière équation permet de déduire les expressions des gains équivalents proportionnel et intégral, données dans la formule (2.18).

b- Nature des approximations

Une approximation exacte avec une loi linéaire peut être obtenue, en dehors des saturations et des points caractéristiques, par une réduction notable de la granularité floue (par exemple dans le cas d'un nombre d'ensembles flous infini, c'est à dire dans le cas limite d'une régulation numérique), ou si certains mécanismes d'interpolation sont choisis. Par exemple, un régulateur flou avec des sorties codées par des singletons et des opérateurs d'agrégation des prémisses et des règles particuliers (respectivement le *produit algébrique* et la *somme bornée*), conduit à une équivalence exacte. L'annexe (1) propose la démonstration de l'obtention d'une loi de régulation floue exactement linéaire dans ce dernier cas.

Les équivalences peuvent servir de base au réglage des RF afin d'apporter des garanties aux utilisateurs: le fait d'initialiser les paramètres du RF, par analogie à des lois de type PID linéaires, peut conduire plus facilement à un comportement final stable. L'avantage des RF ne réside pas, bien sûr, dans la possibilité de générer une loi strictement linéaire. L'intérêt fondamental des RF réside dans la possibilité d'introduire des non-linéarités marquées:

- *Un des intérêts des RF consiste en une mise en oeuvre facile des saturations.* Les saturations naturelles sont forcément utilisées dans la loi de régulation, si le parcours de toute la table est garanti au cours des simulations effectuées. Le fait de calibrer e_{max} et Δe_{max} en tenant compte de l'excursion maximale possible des signaux d'entrée conduit à une loi de type PI saturée, le facteur Δu_{max} permettant de régler le niveau de saturation.
- *Un autre des intérêts des RF consiste à jouer sur les non-linéarités introduites par l'inférence, sur les interpolations naturelles lorsque peu d'ensembles flous, donc de règles, sont utilisés, [FOU,92].* En effet, les interpolations, effectuées entre les points caractéristiques définis plus haut, peuvent être non linéaires, le principe d'équivalence n'étant strictement valide que pour les points caractéristiques de l'espace d'entrée $(e, \Delta e)$. Ces effets non linéaires peuvent, dans certains cas, permettre d'améliorer les performances dynamiques de systèmes linéaires ou non linéaires, selon les conditions de fonctionnement; nous observerons d'ailleurs cela dans le paragraphe (3.2) du chapitre (3). Les paramètres qui influencent principalement sur le type d'interpolation réalisé sont les ensembles flous, les opérateurs mathématiques pour inférer de la connaissance et la méthode de défuzzification.
- Dans le cas où le nombre d'EF, donc de règles, augmente, les effets des interpolations sont de plus en plus locales. Alors, *des non-linéarités intéressantes peuvent être introduites par modification de la base de règles*; nous appliquerons cette approche dans le chapitre (6).

2.2.4 Paramètres du moteur d'inférence

Les choix des paramètres du moteur d'inférence, à savoir les opérateurs de traitement logique de la connaissance, sont multiples. Ci-dessous sont exprimées des indications et informations pour éclairer les choix possibles.

A- Choix de l'opérateur d'agrégation des antécédents

L'opérateur *et* est interprété mathématiquement par un produit cartésien, qui peut se traduire par un *minimum* ou un *produit algébrique*. L'opérateur produit algébrique permet de prendre en compte l'influence de chaque variable et pas seulement l'influence de la variable prédominante.

B- Choix de l'implication et de la règle d'inférence

De nombreuses études en simulation ont été faites pour déduire les fonctions d'implication floue les mieux adaptées à un problème donné. L'idée consiste à trouver, pour une règle compositionnelle d'inférence fixée, les implications floues qui vérifient au mieux

la table logique de critères intuitifs (2.1) -paragraphe (2.1.1)-. Une étude comparative est présentée dans [LEE,90]. Le raisonnement avant est appliqué avec l'opérateur d'inférence *sup_min* à partir de la règle *si x est A alors y est B* pour différentes valeurs floues de *x* qui sont: *très A, plus ou moins A, pas A*. Les résultats proposés sont que les conjonctions de *Mamdani et Larsen* sont bien adaptées pour la régulation floue, malgré une structure logique qui n'est pas bien définie. Des études supplémentaires restent nécessaires. Il s'agit de trouver des règles efficaces pour la régulation floue et bien fondées logiquement; les règles graduelles pourraient être intéressantes, [DUB,91].

C- Choix de l'opérateur d'agrégation des règles

Le choix de l'opérateur d'agrégation des règles doit permettre d'obtenir une relation globale pour la base de règles, avec pour bornes majorante et minorante, respectivement l'union des représentations conjonctives et l'intersection des modèles implicatifs. Les opérateurs co-normes triangulaires sont alors utilisés en association avec des conjonctions et les opérateurs normes triangulaires en association avec les implications pures, [STA,87]. L'opérateur d'agrégation des règles dépend donc de l'interprétation logique choisie pour une règle.

Les règles compositionnelles d'inférence *sup_min* et *sup_produit* étant distributives avec l'opérateur *max* interprétant l'agrégation, il est possible de considérer les différentes règles dans un ordre indifférent. La formule (2.19) résume cette propriété, [LEE,90].

$$(A', B') \circ (\cup_{ij} R_{ij}) = \cup_{ij} [(A', B') \circ (A_i \text{ et } B_j \rightarrow C_k)] \quad (2.19)$$

2.2.5 Bilan: ordre de réglage des paramètres du RF

Ce paragraphe a pour objectif de faire un bilan méthodologique sur la démarche générale de conception d'un RF. Le nombre de paramètres à régler est important, mais une division du processus de conception, accompagnée de réglages initiaux standards pour certains paramètres, facilite la mise au point d'un RF. Les opérations de réglage doivent être appliquées par ordre décroissant de l'influence des paramètres sur les performances et la stabilité en boucle fermée. Les facteurs affichant l'influence la plus globale sont les bornes des univers de discours; le régulateur flou est également, évidemment, particulièrement sensible au contenu de la base de règles.

1. Tout d'abord, une structure a priori de RF est choisie avec des valeurs initiales pour pratiquement tous les paramètres (méthodes de fuzzification et défuzzification, définition des ensembles flous, opérateurs mathématiques pour l'inférence, base de règles initiale).

2. La première opération de calibrage concerne les facteurs d'échelle caractérisant les bornes des univers de discours des trois signaux. L'objectif de cette étape est de fixer la dynamique générale du RF. Cette initialisation peut être faite par analogie avec un régulateur analytique grâce aux relations d'équivalence, ou par essais successifs, comme indiqué dans le paragraphe (2.2.3). Quand le comportement en boucle fermée est stable et que les performances correspondent aux spécifications globales de l'utilisateur, des réglages destinés à optimiser les performances dynamiques peuvent alors être menés localement.
3. L'adaptation de la base de règles peut débuter. La connaissance linguistique a été initialisée, lors de la première étape, à partir d'informations expertes, d'une analyse qualitative, ou d'une phase d'apprentissage à partir de données expérimentales, comme décrit dans le paragraphe (2.2.2). Des non-linéarités locales peuvent alors être introduites dans certaines régions qualitatives du plan de phase, modifiant ainsi la forme de la loi de régulation. Cette étape est réalisable, hors ligne lors d'une étape d'auto-réglage, ou en ligne.
4. Lorsque les performances dynamiques ne peuvent plus être améliorées notablement, des ajustements locaux peuvent parfois permettre d'affiner encore la loi de régulation. Il peut s'agir de modifier légèrement les fonctions d'appartenance des variables d'entrée, les opérateurs d'inférence ou de défuzzification afin d'agir sur les interpolations locales non linéaires engendrées par le processus d'inférence -ce réglage a d'autant plus d'importance que le nombre de règles est faible-

2.3 Aspects théoriques et pratiques de mise en oeuvre

2.3.1 Les problèmes de stabilité et robustesse

A- Stabilité

Rappelons qu'un système est stable s'il atteint un état d'équilibre après avoir subi des perturbations sur son état. Un certain nombre d'outils sont disponibles pour étudier la stabilité des systèmes linéaires ou non linéaires.

Dans le cas de systèmes linéaires invariants, les pôles du système donnent l'information principale sur le comportement dynamique du système et donc, entre autre, sur sa stabilité. Des conditions nécessaires et suffisantes de stabilité peuvent être obtenues à partir d'approches systématiques.

Pour les systèmes non linéaires, les concepts de point d'équilibre, de leurs bassins

d'attraction associés et de cycles limites permettent de caractériser la stabilité. Cependant une approche systématique et unifiée n'est pas disponible. Il est possible de citer, parmi les approches les plus utilisées, la méthode de Lyapunov qui permet de prouver la stabilité asymptotique. La méthode du premier harmonique donne de bonnes indications sur la stabilité après la phase de conception; elle repose sur une approximation des non-linéarités au premier harmonique et sur une analyse a posteriori.

Les régulateurs flous génèrent des lois de commande non linéaires, dont l'expression mathématique est souvent inconnue; en conséquence, les approches suivantes sont utilisables, [MAR,94]:

1. La détermination directe d'une fonction de *Lyapunov* est difficile. Cependant, il est possible de déterminer une telle fonction pour un système bouclé à partir d'une connaissance "pauvre" sur la surface de commande. Cette approche est connue sous le nom de problème de *Luré*, [GAR,94]. Les résultats disponibles sont le critère des petits gains, du cercle, de *Popov*... L'application de ces critères concerne essentiellement les systèmes linéaires. Les non-linéarités éventuelles du système à réguler doivent être traitées comme des incertitudes.
2. La modélisation selon le méthode de *Tagaki-Sugeno* permet d'obtenir des propriétés de stabilité de manière moins conservatrice que précédemment, en s'appuyant sur des fonctions de *Lyapunov* quadratiques, [MAR,95].
3. Des garanties de stabilité pouvant être difficilement intégrées lors de la conception de l'organe de commande, des indications peuvent être données lors d'une étude a posteriori de la stabilité. Une approche possible est la technique du premier harmonique, [MAR,95] [RUG,94]. L'information fréquentielle, obtenue par des expérimentations, permet de détecter les cycles limites et donne des indications sur les changements de gains souhaitables pour les éviter.
4. Une voie de recherche intéressante, en cours d'exploration, est la manipulation de modèles flous pour le système et le régulateur, permettant ainsi de définir de nouveaux critères et de nouveaux outils de stabilité, [GUP,86]. *Braae & Rutherford* ont proposé une méthode, qui consiste à décomposer l'espace d'état en cellules disjonctives floues et à établir la liste des règles permettant de converger vers chaque point d'équilibre. Cette méthode dite *cell to cell mapping* est heuristique et donne des indications de stabilité a posteriori.

B- Les aspects robustesse

Le régulateur étant réglé, initialement et définitivement, à partir du système nominal, des perturbations internes ou externes peuvent être appliquées. D'une part, il est important de connaître les conditions limites de perturbations qui conduisent à la limite de stabilité (*robustesse en stabilité*). D'autre part, il est intéressant de mesurer les limites de perturbations qui entraîne une non-adéquation des performances dynamiques avec le cahier des charges (*robustesse en performances*).

Dans le cadre de ce travail, deux formes d'évaluation de robustesse ont été effectuées. Concernant la robustesse en stabilité, une étude expérimentale a été menée sur le pendule inversé. Les conditions de perturbations limites internes (masse et poids de la tige) ou externes (niveau d'atténuation électrique des signaux de mesure ou de commande), conduisant à la déstabilisation de la tige du pendule, ont été mesurées. Concernant la robustesse en performances, une première évaluation a été menée par des mesures de sensibilité de régulateurs sur les modèles de fours. Les mesures de dégradation des performances, pour une perturbation donnée, sont faites par un indice de sensibilité donné dans la formule (2.20); p_1 et p_2 sont les valeurs d'une performance donnée, évaluées respectivement pour les régulations des systèmes perturbé et nominal.

$$I = \frac{100 (p_2 - p_1)}{p_2} \quad (2.20)$$

2.3.2 Solutions proposées pour les problèmes d'imprécision

Nous mettons ici en évidence certains problèmes pratiques de mise en oeuvre des RF, à savoir la réalisation des inférences en ligne ou hors ligne et la garantie d'une certaine précision de la loi de régulation numérique.

A- Précision d'un PD flou

Des problèmes d'imprécision peuvent apparaître avec un régulateur flou de type PD lors de son utilisation avec inférence en ligne, surtout en régime permanent. En effet, lorsque l'état du système régulé est proche de l'origine du plan de phase, la largeur de base de l'ensemble flou *zéro* conduit à des imprécisions dans le calcul du signal de commande. Pour remédier à cet inconvénient, il est nécessaire de modifier la granularité floue autour de l'origine des univers de discours; pour cela, deux solutions sont envisageables.

Une approche consiste à définir plusieurs niveaux de réglage successifs. Pour des valeurs de l'erreur et de sa variation appartenant au support de l'ensemble flou *ZE*, des commutations sont opérées vers un RF possédant des bornes des univers de discours modifiées. Plusieurs zooms successifs peuvent être opérés pour affiner le réglage autour de l'origine en régime permanent, voir la figure (2.11).

Une autre approche consiste à modifier le nombre ou/et la répartition des ensembles flous des entrées autour de l'origine. Une disposition non linéaire, plus resserrée avec un plus grand nombre d'ensembles flous près de l'origine, conduirait à un réglage plus nuancé et plus précis en régime permanent. Les désavantages de cette méthode sont l'impossibilité de raisonner avec les principes d'équivalence et l'augmentation du nombre d'ensembles flous, donc de règles à développer.

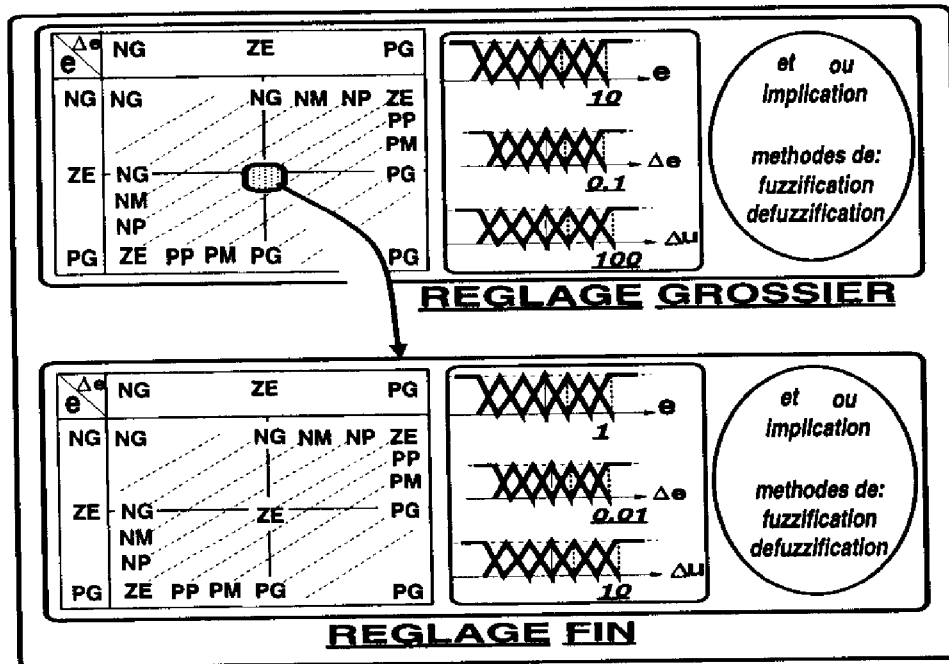


Figure 2.11 : RF avec plusieurs niveaux de réglage.

B- Inférence en ligne ou hors ligne

L'inférence d'une valeur de commande peut être réalisée en ligne, à un niveau logiciel ou matériel. Le fait d'utiliser un microprocesseur dédié permet d'obtenir un organe de commande plus rapide. Malgré le caractère plus lent d'une régulation floue logicielle, il est important de noter que le temps de calcul du RF est indépendant de la complexité des non-linéarités introduites dans la loi de régulation, compte tenu de la nature des inférences. Une autre approche pour réaliser des calculs très rapides durant l'application temps réel consiste à réaliser l'inférence hors ligne. Dans ce cas, les résultats inférés, correspondant à des valeurs pré-fixées des signaux d'entrée du régulateur, sont compilés dans une table de décision numérique. Ces trois approches sont résumées sur les figures (2.12) et (2.13).

Regardons plus en détail le processus de construction hors ligne du RF, sous forme de table d'une décision. La table de décision numérique relie certaines valeurs normalisées des trois signaux e , Δe et u ou Δu . La première étape de la génération hors ligne de la table de régulation numérique consiste à choisir les niveaux de quantification des variables d'entrée, correspondant aux index d'entrée de la table de décision et pour lesquels des valeurs de commande seront inférées. Les ensembles flous caractérisant les différents termes linguistiques sont alors définis sous forme discrète. A chaque niveau de quantification est associé un degré d'appartenance à l'ensemble flou considéré. La fonction d'appartenance d'un EF est caractérisée par un vecteur de dimension égal au niveau de quantification, voir le tableau (2.2).

segment	PS	PM
(0,1)	0,9	0,2
(1,2)	0,2	0,9

Tableau 2.2 : Définition discrète (EF).

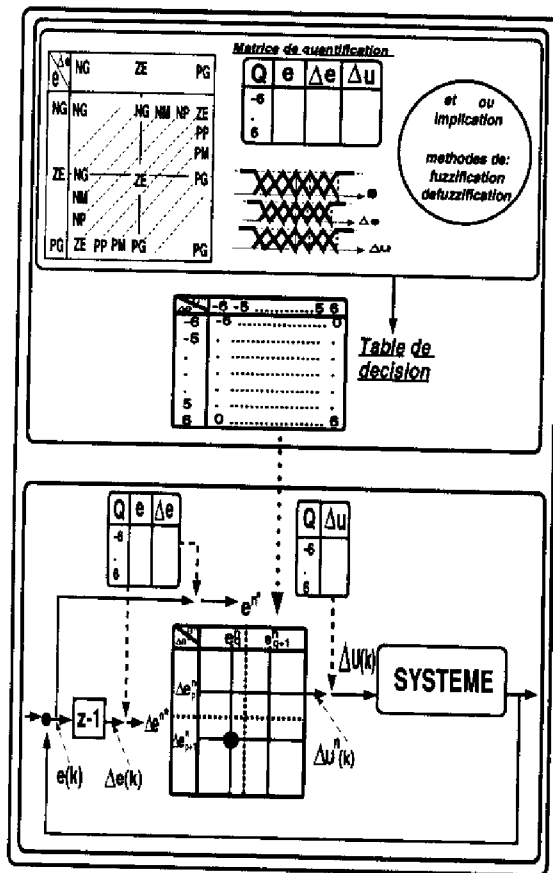


Figure 2.12 : Inférence hors ligne.

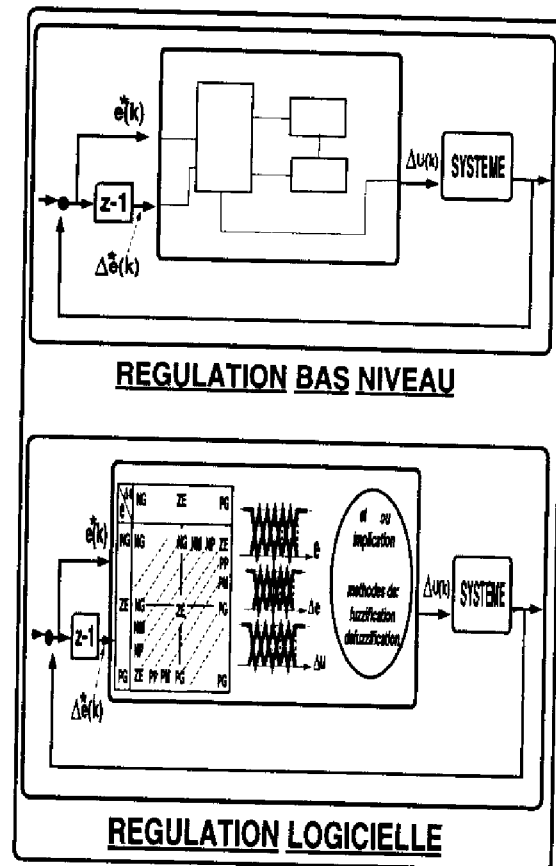


Figure 2.13 : Inférence en ligne.

Les paramètres de la base de définitions des ensembles flous du RF peuvent être regroupés dans la *matrice de quantification*, qui est composée de plusieurs colonnes avec: les niveaux de quantification définis sur les univers de discours normalisés, les valeurs réelles correspondantes des signaux d'entrée ou de sortie du régulateur, les labels des ensembles flous centrés sur ces valeurs, [KET,92]. La loi de régulation numérique peut effectivement être approximée avec les données de la matrice de quantification, la base de règles et les opérateurs mathématiques définis pour le mécanisme d'inférence. Un régulateur flou statique est alors construit, mettant en liaison chaque couple pré-fixé

de valeurs discrètes des variables d'entrée avec la commande numérique correspondante.

Lors de la phase de régulation en ligne avec table de décision, se pose le problème de décisions lorsque des valeurs d'entrées, non traitées lors de l'inférence hors ligne, se présentent. La régulation avec la table de décision numérique consiste donc, d'abord, en une quantification des signaux d'entrée du régulateur, de façon à ce que les valeurs digitales résultantes correspondent aux index d'entrée définis dans la table de décision. Une stratégie habituelle revient à considérer le couple $(e^n, \Delta e^n)$ le plus proche compte tenu d'une distance, comme par exemple la distance euclidienne minimale pour chacune des entrées. Après le passage par le module de quantification, la table de décision numérique permet de générer un niveau de commande à appliquer au système.

Le problème principal posé par la mise en oeuvre du RF avec table de décision est l'apparition du compromis *précision/rapidité* de la régulation. La précision des résultats est en effet réduite par rapport au cas d'une inférence en ligne, à cause du phénomène de quantification des signaux d'entrée, qui entraîne une approximation de la loi de régulation.

C- Régulation précise avec table de décision

La quantification au niveau le plus proche, pour une valeur d'entrée donnée, génère des non-linéarités sous forme de discontinuités. Ces effets sont particulièrement sensibles lors des régimes permanents pour lesquels une plus grande précision est requise. Des moyens permettent d'améliorer la précision de détermination des valeurs de commande, de se rapprocher des résultats d'inférence en ligne, tout en conservant une méthode d'utilisation rapide du régulateur flou en ligne.

Le premier moyen consiste à modifier certains paramètres utilisés pour la génération du régulateur flou hors ligne. Il est en effet possible d'augmenter le nombre de niveaux de quantification utilisés pour la construction de la table de décision. Toutefois, la zone mémoire de stockage de la table de décision s'en trouve accrue.

Des méthodes intermédiaires, moins coûteuses en espace mémoire, existent. On peut envisager de commuter entre deux tables de décision (ou plus) lors de la régulation en ligne. Une *régulation grossière* est appliquée pendant le régime transitoire et une *régulation fine* prend le relais pendant le régime permanent. L'idée est de minimiser les erreurs de quantification autour de l'origine où une plus grande précision est nécessaire, le caractère approximatif étant moins pénalisant loin de la référence. Les deux réglages *fin* et *grossier* diffèrent principalement par les paramètres de la matrice de quantification. Une quantification deux fois plus précise est utilisée au cours du régime de régulation le plus délicat, sans avoir besoin d'agrandir inutilement l'espace mémoire utilisé.

Nous proposons une autre idée, consistant à pondérer plusieurs valeurs extraites de la table de décision. Durant la phase de régulation, les quatre valeurs adjacentes,

encadrant les entrées θ^{n*} et $\Delta\theta^{n*}$, sont lues dans la table de décision et une interpolation permet de déduire une valeur à envoyer au système, voir l'équation (2.21) et la figure (2.14).

$$U_n(k) = \frac{\sum_{i=q}^{q+1} \sum_{j=p}^{p+1} U(i, j) \times A(i, j)}{\sum_{i=q}^{q+1} \sum_{j=p}^{p+1} A(i, j)} \quad (2.21)$$

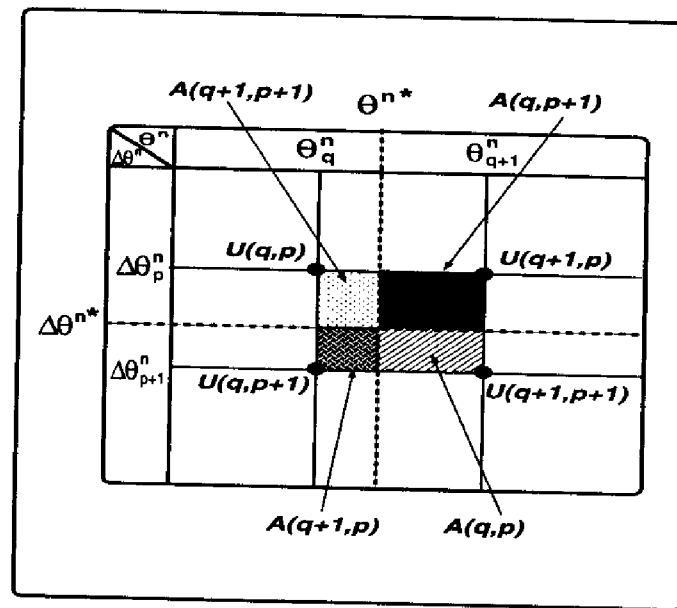


Figure 2.14 : Illustration graphique de l'interpolation.

Après avoir détaillé les spécificités des RF, nous allons maintenant présenter des résultats de simulations et d'expériences.

Chapitre 3

Régulation floue: Applications

Nous présentons ici des expérimentations avec un régulateur flou de la première génération. Les essais ont été menés sur certains modèles du pendule inversé et sur les fours simulés. Ils sont destinés à mettre à jour les mécanismes de génération d'une loi de commande numérique, conduisant à des améliorations de performances par rapport à certaines régulations conventionnelles.

Le paragraphe (3.1) expose les résultats d'expériences menées sur différents modèles du pendule inversé (modèles linéaire et non linéaire avec deux variables d'état, avec, dans le second cas, un départ pour la tige de la position angulaire basse) et sur le pendule réel. Le paragraphe (3.2) concerne des applications sur des modèles de fours. Les conclusions du chapitre sont fournies en (3.3).

3.1 Applications sur le pendule inversé

Une première évaluation des techniques floues de base a été faite sur le pendule inversé, en menant des études comparatives avec des régulateurs algorithmiques (régulateurs à retour d'état, en cascade, PID et à placement de pôles), pour différents modèles du pendule, voir le paragraphe (1.4.1) du chapitre (1), [DEG,93] [DEG,94a] [DEG,95a].

3.1.1 Modèle linéaire, comparaison avec la littérature

L'objet de ces premières simulations est de comparer les résultats obtenus avec un RF réglé heuristiquement et deux régulateurs linéaires. Un modèle linéaire du pendule est utilisé. L'objectif est d'observer expérimentalement la capacité du RF à améliorer certains résultats de régulations conventionnelles. Bien sûr, un régulateur mathématique linéaire utilisant un modèle analytique du système est forcément plus précis qu'un RF; toutefois, son aspect linéaire lui confère peu de degrés de liberté pour optimiser ses performances. Les améliorations observées seront expliquées ultérieurement.

A- Correcteurs considérés

Les correcteurs conventionnels considérés sont utilisés dans [THE,89]. Le problème posé est de concevoir des organes de régulation, qui permettent de garantir un temps de réponse à $\epsilon\%$ ($\epsilon \simeq 0$) de 3s en boucle fermée. La condition angulaire initiale de 0.1 rad appartient au domaine de fonctionnement linéaire. Le premier régulateur à retour d'état (RE) est défini par l'équation (3.1). Le comportement en boucle fermée est fixé par une fonction de transfert du second ordre de constante de temps double 0.5s (le temps de réponse d'un premier ordre à 5% est en effet égal à trois fois la constante de temps). Le deuxième régulateur en cascade (RC) est défini par l'équation (3.2); dans ce cas, aucune information sur la vitesse angulaire, assez difficile à mesurer, n'est utilisée. Le comportement en boucle fermée est fixé par une fonction de transfert du troisième ordre de constante de temps triple $\frac{1}{3}$ s.

$$u = \theta^{consigne} - [r_1 \ r_2] [\theta \ \dot{\theta}]^T \quad (3.1)$$

$$u = K \frac{1 + cp}{1 + dp} \theta \quad (3.2)$$

Compte tenu de l'ordre du système, la structure de l'organe de commande est de type PD. Le régulateur flou mis en place est un régulateur de *Mamdani*. Les paramètres suivants sont utilisés: la base de règles de *MacVicar-Whelan* (dérivée d'une analyse qualitative), sept EF triangulaires uniformément répartis (pour chaque variable), le *minimum* pour exprimer le *et*, la conjonction de *Mamdani*, l'opérateur

maximum pour l'agrégation des règles, la règle compositionnelle d'inférence de *Zadeh*, la méthode de défuzzification par moyenne pondérée, la fuzzification en singletons. La loi de normalisation est choisie linéaire; les bornes des univers de discours, θ_{max} , $\Delta\theta_{max}$ et u_{max} sont réglées avec la procédure heuristique du paragraphe (2.2.3) du chapitre(2). Le résultat de cette étape a conduit aux valeurs: $\theta_{max} = 0.1rad$, $\Delta\theta_{max} = 0.21rad/s$ et $u_{max} = 25N$.

Une stratégie de régulation avec table de décision a été choisie; le nombre de niveaux de quantification est 25. La régulation est mise en oeuvre avec trois niveaux de réglages fins successifs, ceci afin de pallier les erreurs en régime permanent. Chaque réglage est défini automatiquement en fonction du précédent, par un facteur de proportionnalité de $\frac{1}{5}$ par rapport aux bornes des univers de discours, servant également à définir la condition de commutation sur θ et $\Delta\theta$.

B- Type de modèle et de simulation numérique utilisés

Pour les besoins des comparaisons effectuées, il est nécessaire de préciser la nature exacte du modèle utilisé et les conditions de simulation numérique. Le même modèle que dans [THE,89] est considéré, à savoir un modèle linéaire du pendule avec deux variables d'état, θ et $\dot{\theta}$ -voir le paragraphe (1.4.1- chapitre (1)), avec les paramètres: $l = 1.2m$, $m = 2Kg$, $M = 10Kg$. La simulation a été menée avec une constante d'échantillonnage de 10ms, en accord avec le théorème de *Shannon* et connaissant l'expression de la constante de temps du modèle linéaire ($T = \frac{1}{\sqrt{\frac{3g}{l} \frac{M+m}{7M+4m}}}$).

Le régulateur n'est connu que par ses entrées-sorties numériques, des simulations avec un modèle numérisé du système sont donc nécessaires. Nous avons utilisé un système d'équations différentielles linéaires, intégré par une procédure de type *Runge Kutta*. La dynamique est un peu moins bien traduite que dans [THE,89], mais c'est l'approche qui permet de se rapprocher au mieux de la situation de simulation continue, développée dans cette référence.

C- Résultats et conclusions:

Le tableau (3.1) résume l'étude comparative du comportement du pendule pour rallier la position verticale à partir d'une condition initiale de 0.1rad. D représente le pourcentage de dépassement, $\dot{\theta}_{max}$ la vitesse angulaire maximale et U_{max} la force maximale appliquée. Le régulateur flou fournit une réponse plus rapide que les autres régulateurs linéaires, avec un dépassement très faible, un coût de commande faible et une vitesse angulaire maximale légèrement supérieure à celle obtenue avec le régulateur à retour d'état. Le RF offre donc un moyen de doser les commandes différemment, afin de mieux atteindre certaines spécifications en boucle fermée. En début de régime transitoire, les commandes décroissent moins vite que celles du régulateur à réaction d'état.

Les résultats comparatifs doivent être nuancés, par le fait que le type de simulation numérique effectué ne traduit pas la dynamique du système exactement de la même façon que dans [THE,89].

	$t_{r,\epsilon\%}$ s	D %	θ_{max} rad/s	u_{max} N
RE	3.7	0	0.075	25
RC	3.25	70	-	100
RF	3	0.2	0.08	25

Tableau 3.1 : Comparaison des performances du RF et des régulateurs linéaires.

De petites améliorations ont pu être détectées; le problème consiste maintenant à en analyser de plus près les causes. Des expériences comparatives avec un PD analytique vont permettre de caractériser les particularités de la loi de régulation floue: les différences sont-elles dues aux interpolations non linéaires, aux saturations, à la quantification,.... ?

3.1.2 Modèle non linéaire, analyse de la loi de régulation

Un modèle plus proche de la réalité est maintenant utilisé; il s'agit du modèle non linéaire avec deux variables d'état -voir le paragraphe (1.4.1) chapitre (1)-. Les valeurs des paramètres ont été choisies égales à: $l = 0.26m$, $m = 0.37Kg$, $M = 1.4Kg$, en liaison avec le système réel. L'étude d'un RF de type PD, mis en oeuvre de différentes façons, met en évidence l'intérêt d'une simple loi de régulation floue par rapport à une loi PD analytique équivalente donnée en (3.3).

A- Démarche suivie

L'objectif des premières expériences est de réguler la tige, pour une condition angulaire initiale de $-0.4rad$. Les spécifications en boucle fermée sont une convergence la plus rapide possible, avec un temps de réponse à $\epsilon\%$ minimal (ϵ très petit) et un dépassement nul.

Le régulateur flou choisi est de type *Mamdani* et possède les mêmes paramètres que dans le paragraphe précédent. Il est réglé expérimentalement de manière à atteindre les spécifications. Ensuite le régulateur PD analytique équivalent est construit suivant les règles d'équivalence (3.4). La régulation PD analytique, saturée à 100 et permettant d'atteindre les spécifications, est présentée sur les figures (3.1) et (3.2). Les paramètres du PD sont $(K_p, K_d) = (\frac{100}{0.4}, \frac{100}{3.6})$. Les performances dynamiques atteintes sont $[tr_{\epsilon\%}, D, max(u), max(\Delta\theta)] = (0.47s, 0, 100N, 2.5rad/s)$.

$$u(k) = K_p \theta(k) + K_d \Delta\theta(k) \quad (3.3) \quad K_p = \frac{u_{max}}{\theta_{max}} \quad ; \quad K_d = \frac{u_{max}}{\Delta\theta_{max}} \quad (3.4)$$

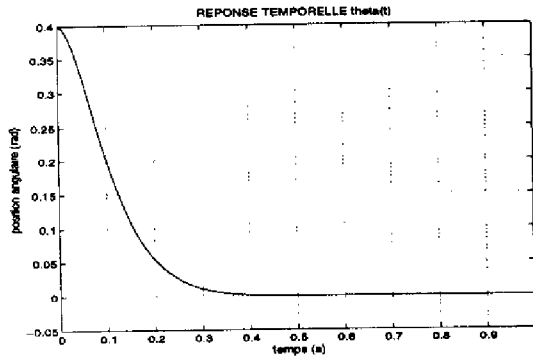


Figure 3.1 : Régulateur PD, θ .

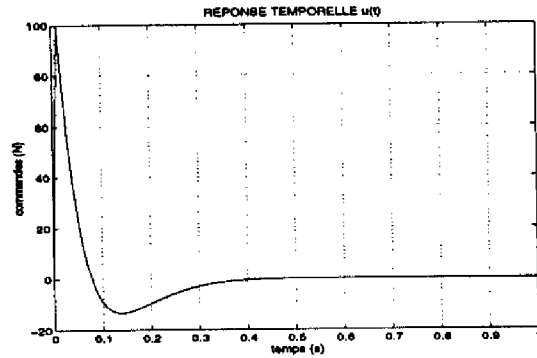


Figure 3.2 : Régulateur PD, U .

Notre souci est de mettre en oeuvre, au mieux, la loi de régulation floue, calculée hors ligne. Nous désirons, également, souligner un des intérêts de cette simple loi de régulation floue par rapport à son quasi-équivalent linéaire. Concernant ce second point, il ne s'agit pas de mettre en évidence l'intérêt de l'introduction naturelle de saturations dans la loi de régulation floue, la loi analytique étudiée possédant également une saturation sur la commande. Il s'agit d'observer l'influence, sur les performances dynamiques en boucle fermée, des interpolations non linéaires locales liées aux calculs de l'inférence pour le RF. Toutefois, avec une mise en oeuvre avec table de décision, le phénomène de quantification des univers de discours manipulés interfère avec les effets des non-linéarités locales de la loi de régulation floue. Le premier problème posé consiste alors à utiliser un mécanisme simple pour générer des commandes en ligne, par un calcul rapide et un rapprochement suffisant avec la loi de régulation floue E-S non discrétisée.

B- Régulation floue avec une table de décision

La première mise en oeuvre du RF consiste à utiliser directement la table de décision avec le minimum de niveaux de quantification, de façon à minimiser l'espace mémoire nécessaire. 13 puis 25 niveaux de quantification ont été considérés, voir la figure (3.3).

Pour 13 niveaux, les valeurs de commande inférées correspondent aux points modaux ou aux points de recouvrement $\frac{1}{2}$. Ces points sont ceux pour lesquels l'équivalence avec une loi linéaire PD est exacte; l'équation (3.3) exprime la relation E-S aux niveaux de quantification. On obtient alors une loi linéaire discrétisée avec un pas grossier; la quantification introduit donc un effet non linéaire sous forme de discontinuités.

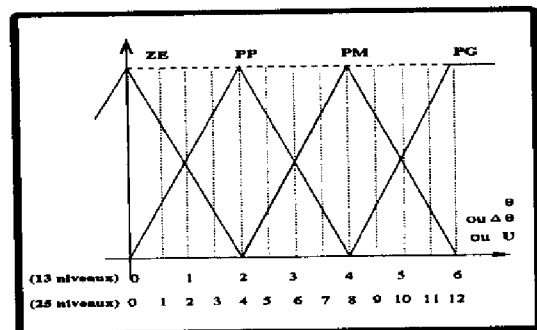


Figure 3.3 : Niveaux de quantification.

Les figures (3.4) et (3.5) présentent le résultat d'une simulation avec les paramètres $(\theta_{max}, \Delta\theta_{max}, u_{max})=(0.4\text{rad}, 3.11\text{rad/s}, 100\text{N})$. Les performances dynamiques atteintes sont $[tr_{\varepsilon\%}, D, \max(u), \max(\Delta\theta)]=(0.5\text{s}, 0.036\text{rad}, 100\text{N}, 2.4\text{rad/s})$. Il est possible d'observer les effets de la quantification grossière. Le signal de commande est très irrégulier. De plus, une imprécision notable apparaît en régime permanent est très irrégulier. De plus, une imprécision notable apparaît en régime permanent sous forme d'oscillations d'amplitude 0.036rad. Cette valeur correspond à la valeur médiane entre les deux premiers niveaux de quantification, donc au cas limite pour lequel une action de commande nulle est inférée.

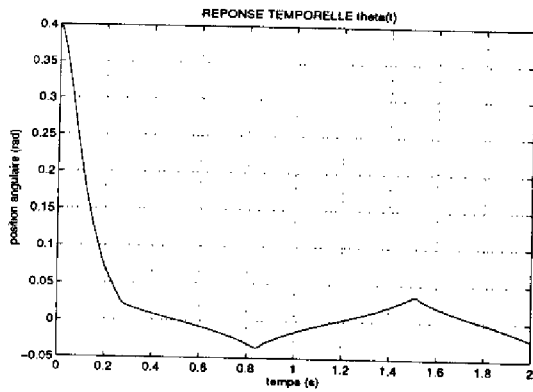


Figure 3.4 : RF (nq= 13), θ .

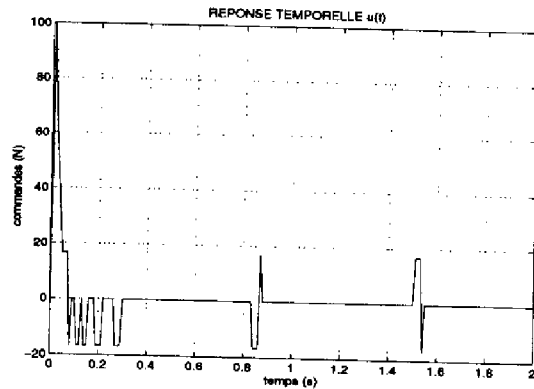


Figure 3.5 : RF (nq= 13), U .

Ensuite, 25 niveaux de quantification sont utilisés. Les figures (3.6) et (3.7) présentent le résultat d'une simulation avec les paramètres $(\theta_{max}, \Delta\theta_{max}, u_{max})=(0.4\text{rad}, 3\text{rad/s}, 100\text{N})$. Les performances dynamiques atteintes sont $[tr_{\varepsilon\%}, D, \max(u), \max(\Delta\theta)]=(0.5\text{s}, 0.018\text{rad}, 100\text{N}, 2.2\text{rad/s})$. Le signal de commande est moins discontinu et la précision de la réponse en régime permanent est multipliée par 2. Les points intermédiaires introduits permettent d'ajouter des niveaux de commande dans les domaines des non-linéarités locales où l'équivalence avec une loi linéaire n'est plus exacte.

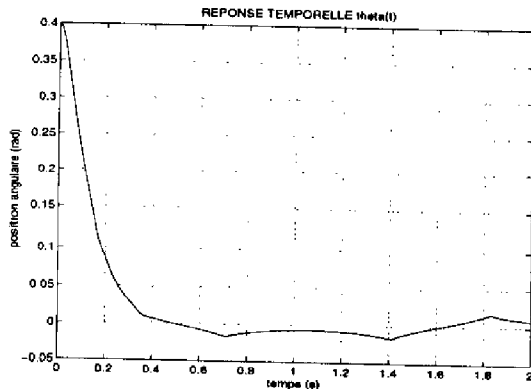


Figure 3.6 : RF (nq= 25), θ .

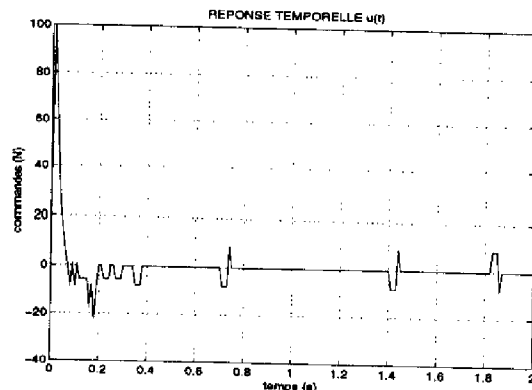


Figure 3.7 : RF (nq= 25), U .

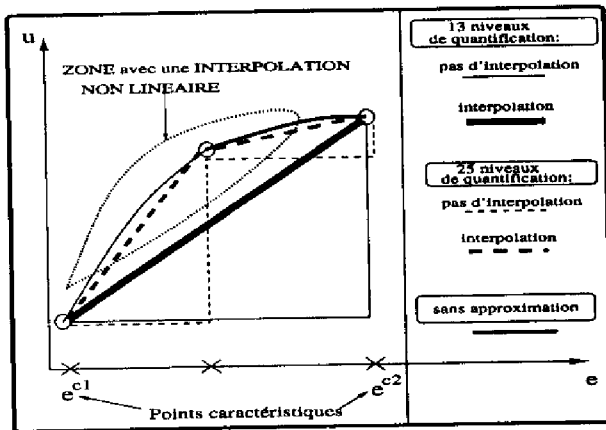


Figure 3.8 : Mise en évidence des effets non linéaires de l'interpolation.

Le choix du nombre de niveaux de quantification permet d'accentuer plus ou moins les non-linéarités introduites par le mécanisme d'inférence, voir la figure (3.8) pour une illustration. Toutefois, il est impossible d'observer l'influence spécifique des non-linéarités liées à l'inférence, leurs conséquences étant trop combinées aux effets non linéaires des quantifications.

Des méthodes intermédiaires sont maintenant recherchées pour masquer le processus de quantification sans avoir à utiliser trop de niveaux de quantification, ce qui conduirait à de grandes tables de décisions.

C- Régulation floue avec plusieurs tables de décision

La première approche consiste à définir plusieurs zones de fonctionnement, correspondant à des zooms successifs, à mesure que les erreurs sur l'angle et sa variation diminuent, comme décrit dans le paragraphe (2.3.2) du chapitre (2). Les figures (3.9) et (3.10) présentent le résultat d'une simulation avec les paramètres $(\theta_{max}, \Delta\theta_{max}, u_{max}) = (0.4\text{rad}, 3.6\text{rad/s}, 100\text{N})$ et 25 niveaux de quantification. Les performances dynamiques atteintes sont $[tr_{\epsilon\%}, D, \max(u), \max(\Delta\theta)] = (0.5\text{s}, 7 \cdot 10^{-4}\text{rad}, 100\text{N}, 2.5\text{rad/s})$.

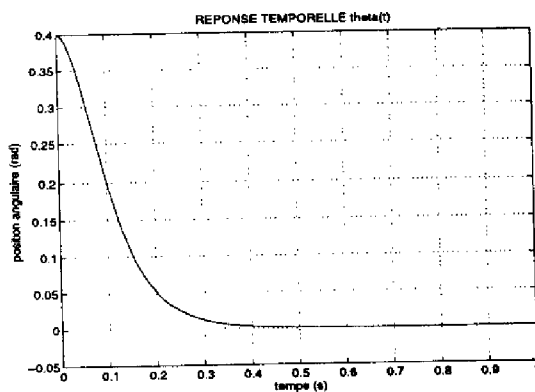


Figure 3.9 : RF (3 tables), θ .

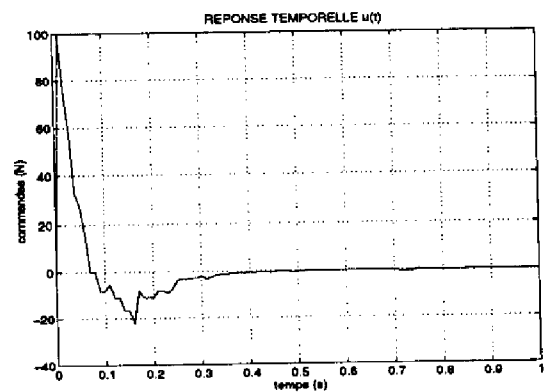


Figure 3.10 : RF (3 tables), U .

Nous pouvons observer une nette amélioration par rapport à un seul niveau de réglage. L'imprécision en régime permanent est fortement réduite et le signal de com-

mande est moins anguleux. La comparaison à la régulation PD analytique montre que le temps de réponse est un peu moins bon et que de petites oscillations sont apparues. Les effets des interpolations non linéaires se combinant aux erreurs d'approximation numérique dues à l'utilisation de niveaux de quantification, il est impossible de dissocier la cause de la dégradation de la réponse de la régulation floue par rapport à la régulation linéaire.

D- Régulation floue avec une table de décision et interpolation en sortie

Une interpolation linéaire entre quatre valeurs extraites de la table de décision est opérée, comme présenté dans le paragraphe (2.3.2) du chapitre (2). Pour que la stratégie adoptée fasse apparaître de façon significative les non-linéarités locales dues au processus d'inférence, il faut que le nombre de niveaux de quantification soit supérieur à une borne minimale pour faire intervenir des points autres que les points caractéristiques. Cette limite est 13. Dans ce dernier cas, l'interpolation entre les points de la table conduit à une loi exactement linéaire.

Le choix d'un nombre de niveaux de quantification égal à 25 suffit à améliorer les résultats de la régulation PD linéaire. Les figures (3.11) et (3.12) présentent le résultat d'une simulation avec les paramètres $(\theta_{max}, \Delta\theta_{max}, u_{max}) = (0.4\text{rad}, 3.6\text{rad/s}, 100\text{N})$. Les performances dynamiques atteintes sont $[tr_{\epsilon\%}, D, \max(u), \max(\Delta\theta)] = (0.35\text{s}, 7 \cdot 10^{-4}\text{rad}, 100\text{N}, 2.5\text{rad/s})$. Un réglage équivalent au PD analytique permet donc d'améliorer le temps de réponse avec un très faible dépassement. Une légère modification du réglage de la dynamique du RF conduit à un dépassement nul, avec un temps de réponse plus élevé, mais toujours meilleur à celui du PD analytique. En effet, les paramètres $(\theta_{max}, \Delta\theta_{max}, u_{max}) = (0.4\text{rad}, 3\text{rad/s}, 100\text{N})$, les performances dynamiques atteintes sont $[tr_{\epsilon\%}, D, \max(u), \max(\Delta\theta)] = (0.39\text{s}, 0\text{rad}, 100\text{N}, 2.5\text{rad/s})$.

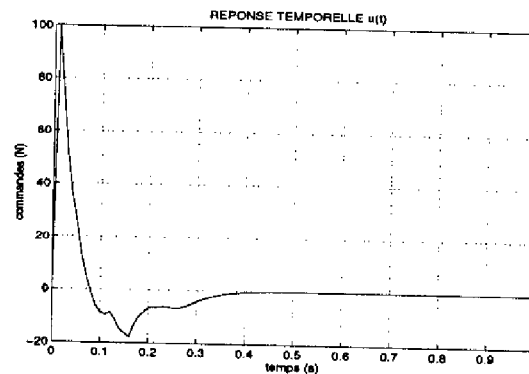
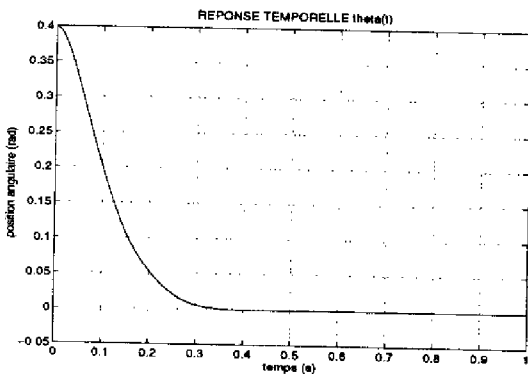


Figure 3.11 : RF (Interpolation), θ .

Figure 3.12 : RF (Interpolation), U .

L'interpolation utilisée, pour déduire une valeur de commande en sortie de la table de décision, minimise les erreurs d'approximation numérique liées au phénomène de quantification. Une méthode pratique de mise en oeuvre de la loi de régulation floue a été proposée; elle est efficace car rapide, simple, nécessitant peu d'espace mémoire et permettant de se rapprocher notablement de la loi de régulation floue avec inférence

en ligne. Les effets non linéaires dus à l'inférence sont alors apparus clairement, dans notre cas particulier, voir la figure (3.8). Les résultats montrent que les non-linéarités locales générées ici permettent d'améliorer les performances dynamiques par rapport à un PD linéaire.

3.1.3 Modèle non linéaire, départ de la position basse

Essayons maintenant de stabiliser la tige du pendule avec une loi de commande floue, en partant d'une condition angulaire initiale plus contraignante. Pour réaliser la stabilisation du pendule à partir de la condition initiale π rad, une loi de commande non linéaire doit être générée. Deux approches principales sont alors possibles, à savoir introduire les non-linéarités par le biais des EF répartis de façon spécifique, ou à travers la base de règles non standard avec des EF uniformément répartis. La deuxième approche a été adoptée ici.

A- Stratégie employée

La structure de commande floue est composée de deux régulateurs. Dans le domaine angulaire $[-0.4, 0.4]$ rad, le RF développé précédemment permet de stabiliser la tige. Pour les autres valeurs angulaires, un régulateur flou mettant en oeuvre une loi de type *tout ou rien* améliorée prend le relai. La figure (3.13) représente le principe de cette structure de commande.

L'intérêt d'utiliser un module flou pour la régulation grand angle réside dans sa capacité à exprimer des nuances qualitatives en fonction des valeurs de $\Delta\theta$. La variable θ peut prendre les valeurs floues *NG*, *ZE* et *PG*, alors que la variable $\Delta\theta$ peut varier de *NG* à *PG* avec cinq labels intermédiaires. Les règles sont établies d'après des considérations physiques. Le problème posé consiste à nuancer les valeurs des commandes en fonction des vitesses d'évolution de la tige. Le principe général de la stratégie de commande est d'aider à poursuivre un mouvement en cours, l'action étant d'autant plus faible que la vitesse déjà acquise est élevée. La discontinuité de l'action de commande lorsque $\Delta\theta$ change de signe est gérée en appliquant des actions nulles pour des vitesses nulles. Pour les angles, dont la valeur absolue est inférieure à $\frac{\pi}{2}$ rad, la tige évolue librement compte tenu de l'énergie acquise dans le cadran précédent.

La figure (3.14) résume la stratégie de régulation aux grands angles. La table de règles déduite est donnée dans le tableau (3.2).

$\theta \downarrow \Delta\theta \rightarrow$	NG	NM	NP	ZE	PP	PM	PG
NG	NP	NM	NG	ZE	PM	PP	ZE
ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE
PG	ZE	NP	NM	ZE	PG	PM	PP

Tableau 3.2 : Règles pour les grands angles.

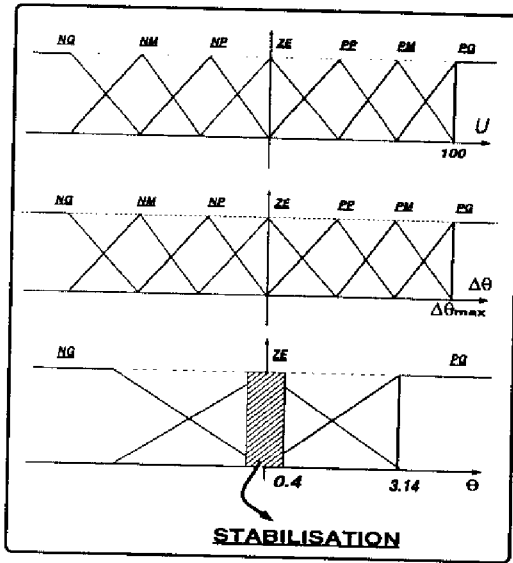


Figure 3.13 : Principe de la structure de commande floue.

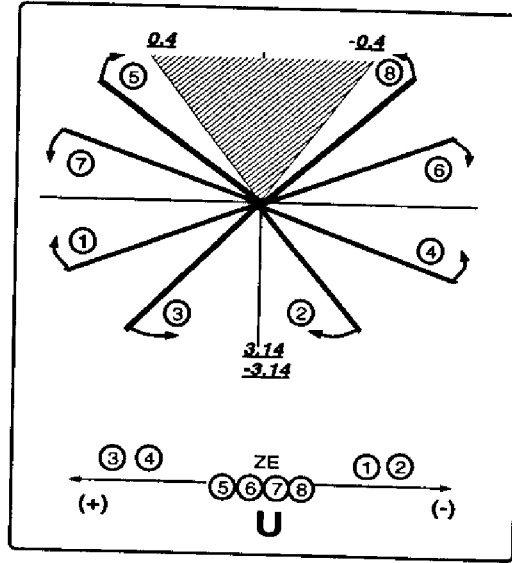


Figure 3.14 : Principe de la stratégie de régulation aux grands angles.

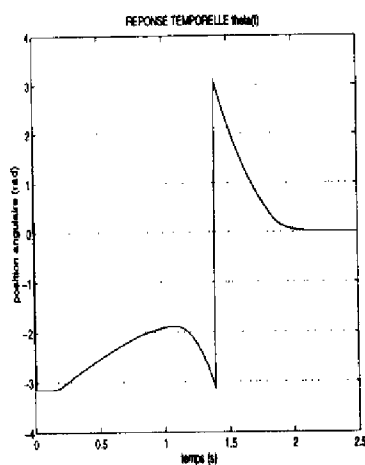
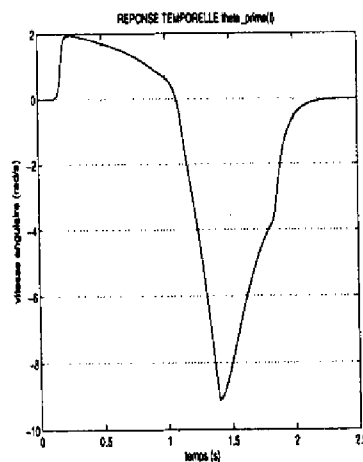
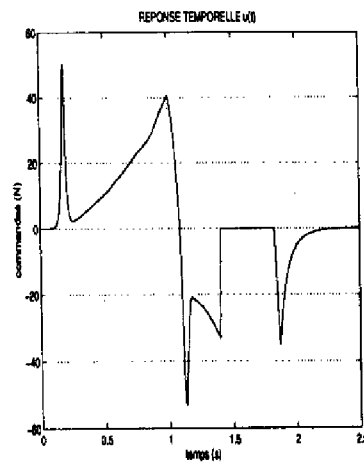
B- Résultats

Le réglage du paramètre $\Delta\theta_{max}$ pour la régulation aux grands angles est fait de manière à moduler l'efficacité du dosage de la commande. Le compromis est d'atteindre la zone de stabilisation le plus rapidement possible, tout en ne transmettant pas trop d'énergie, afin de permettre au régulateur flou stabilisateur de prendre le relai sans dépassement. La valeur a été réglée à 2rad/s, la table de décision inférée hors ligne est donnée sur le tableau (3.3) et les courbes résultats sont fournies en (3.15), (3.16) et (3.17).

$\theta^n \downarrow \Delta\theta^n \rightarrow$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
-6	-2	-3	-4	-5	-6	-3	0	2	4	3	2	1	0
-5	-1.66	-2.25	-3.33	-3.75	-5	-2.25	0	1.5	3.33	2.25	1.66	0.75	0
-4	-1.33	-1.8	-2.66	-3	-4	-1.8	0	1.2	2.66	1.8	1.33	0.6	0
-3	-1	-1.5	-2	-2.5	-3	-1.5	0	1	2	1.5	1	0.5	0
-2	-0.66	-1.2	-1.33	-2	-2	-1.2	0	0.8	1.33	1.2	0.66	0.4	0
-1	-0.33	-0.75	-0.66	-1.25	-1	-0.75	0	0.5	0.66	0.75	0.33	0.25	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	-0.25	-0.33	-0.75	-0.66	-0.5	0	0.75	1	1.25	0.66	0.75	0.33
2	0	-0.4	-0.66	-1.2	-1.33	-0.8	0	1.2	2	2	1.33	1.2	0.66
3	0	-0.5	-1	-1.5	-2	-1	0	1.5	3	2.5	2	1.5	1
4	0	-0.6	-1.33	-1.8	-2.66	-1.2	0	1.8	4	3	2.66	1.8	1.33
5	0	-0.75	-1.66	-2.25	-3.33	-1.5	0	2.25	5	3.75	3.33	2.25	1.66
6	0	-1	-2	-3	-4	-2	0	3	6	5	4	3	2

Tableau 3.3 : Table de décision utilisée pour la régulation en ligne aux grands angles.

Les résultats correspondent à la trajectoire attendue. Etant d'abord au repos et en position basse, des actions de commande positives entraînent la tige vers les angles négatifs ($t \simeq 0.15s$), la vitesse d'évolution de la tige reste faible. Après un premier balancement ($t \simeq 1s$), la tige repasse par la position basse, qui est le point de *cassure* ($\pi, -\pi$) dans la définition de l'angle. La vitesse de convergence vers la position but devient importante, car des commandes négatives renforcent ce mouvement ($t \simeq 1.1s$). A partir de ($t \simeq 1.4s$), la commande est nulle, car la vitesse de convergence de la tige est grande et le système utilise l'énergie stockée. Enfin, dans la zone de régulation aux petits angles (au-dessous de $0.4rad$ à partir de $t \simeq 1.8s$), des commandes négatives sont appliquées afin d'anticiper sur les dépassements, compte tenu des vitesses de convergence importantes.

Figure 3.15 : θ .Figure 3.16 : $\Delta\theta$.Figure 3.17 : U .

3.1.4 Applications sur le système réel

Il est impossible d'utiliser le régulateur flou conçu pour la seule régulation de θ sur le système réel. En effet, les expériences sont rapidement interrompues par le fait que le chariot atteint les limites du rail. En conséquence, un régulateur flou pouvant réguler à la fois θ et y doit être développé. Dans un premier temps, différentes stratégies permettant de mener à bien cette double régulation sont présentées; ensuite, la méthode employée est détaillée; enfin, des résultats expérimentaux sur le système réel sont comparés à ceux d'un régulateur à placement de pôles, [REH,93].

A- Choix d'une stratégie de régulation avec un système à quatre sorties

Une première approche consiste à développer une base de règles avec quatre entrées, ou, de façon équivalente, à écrire une table de règles à deux entrées θ et $\dot{\theta}$ -évalué

numériquement- pour chaque couple (y, \dot{y}) . Cette démarche assure beaucoup de degrés de liberté, mais l'écriture des règles s'avère fastidieuse. Une approche simplifiée revient à combiner les résultats fournis par le traitement en parallèle de deux bases de règles ayant respectivement pour entrées $(\theta, \dot{\theta})$ et (y, \dot{y}) .

Une autre méthode allie deux systèmes flous en série, pour générer les valeurs de commande, voir la figure (3.18). Une première base de règles, dépendante de y et \dot{y} , génère une information qui sert à moduler le traitement de la deuxième base de règles, dépendante de θ et $\dot{\theta}$.

La technique adoptée suit le principe suivant. Elle n'utilise qu'une table de décision, à savoir celle qui a été dérivée pour la régulation de la seule position angulaire dans les paragraphes précédents. L'utilisation de cette table est maintenant modifiée, en ce sens que l'index de θ_n est décalé en fonction des valeurs courantes de y et \dot{y} . Cette approche est de loin la plus facile à mettre en place, par rapport à d'autres mises en oeuvre floues et conventionnelles, [MOR,76].

Le décalage de l'index θ_n à l'entrée de la table de décision, revient à définir, à chaque instant d'échantillonnage, une référence angulaire, $\theta_{ref}(k)$, qui dépend de la position et de la vitesse du chariot. Les formules (3.5) expriment l'indice décalé de la table de décision $\theta_n^*(k)$, en fonction de l'indice initial normalisé $\theta_n(k)$ et de la référence angulaire; les termes $f_{\theta_{no}}$, $f_{y_{no}}$ et $f_{\dot{y}_{no}}$ sont les facteurs de normalisation. Des saturations sont définies heuristiquement sur les facteurs $f_{y_{no}}$, $f_{\dot{y}_{no}}$, afin de ne pas déstabiliser le système, en générant des indices normalisés trop grands pour l'angle. Une position horizontale positive, avec une très faible vitesse du chariot, correspond à une référence angulaire négative. La figure (3.19) montre que, dans le futur, les commandes appliquées tendront à faire converger y et θ vers leur position cible. En effet, à mesure que y diminue, θ_{ref} converge vers 0; des commandes négatives sont alors appliquées, afin que l'angle tende vers sa référence.

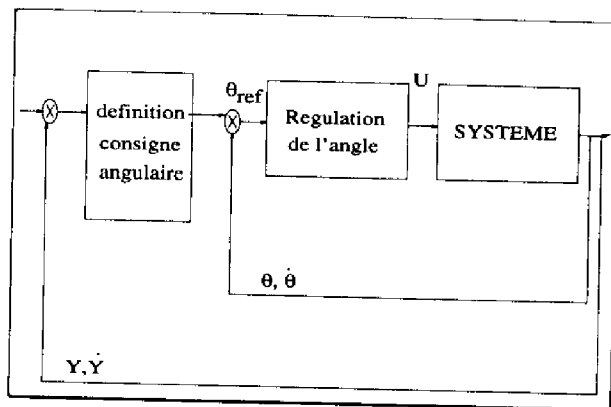


Figure 3.18 : Structure en cascade.

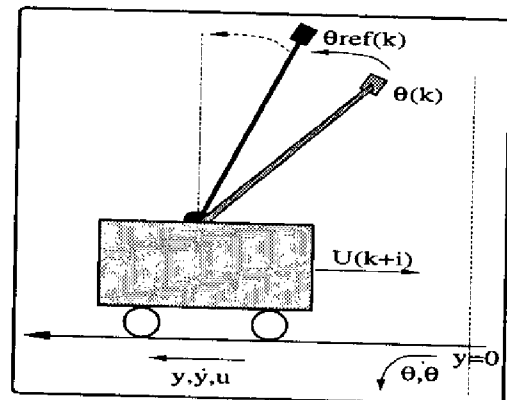


Figure 3.19 : Principe de la référence angulaire.

$$\begin{cases} \theta_n^*(k) = \theta_n(k) - \theta_{ref}(k) \\ \theta_n(k) = f_{\theta_{no}} \theta(k) \\ \theta_{ref}(k) = -\{f_{y_{no}} y(k) + f_{\dot{y}_{no}} \dot{y}(k)\} \end{cases} \quad (3.5)$$

B- Résultats comparatifs à un RPP

Une expérience a été réalisée sur le système réel avec un RF, mis en oeuvre avec une table de décision et la technique d'interpolation. Les valeurs de paramètres suivantes du RF ont été utilisées: $f_{\theta_{no}} = 3$, $f_{\Delta\theta_{no}} = 17.15$, $f_{y_{no}} = 0.5$, $f_{\dot{y}_{no}} = 1.5$ et $u_{max} = 120N$. La référence pour la position du chariot est choisie égale à 53cm. Les signaux capteurs sont filtrés par un premier ordre, afin d'obtenir de bonnes qualités de mesure pour la régulation. Une compensation est appliquée, il s'agit d'une action de commande constante opposée aux frottements de *Coulomb*.

Les résultats sont fournis sur les figures (3.20) à (3.25).

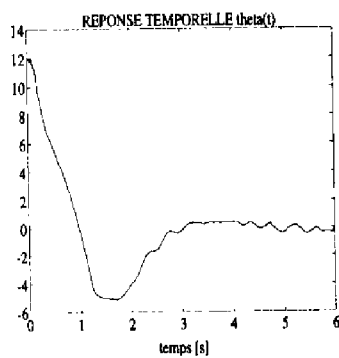


Figure 3.20 : θ .

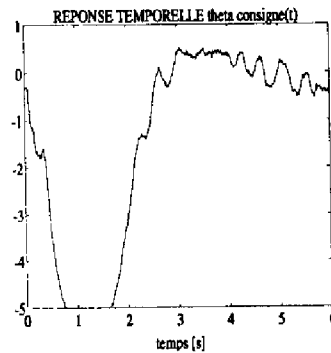


Figure 3.21 : θ_{ref} .

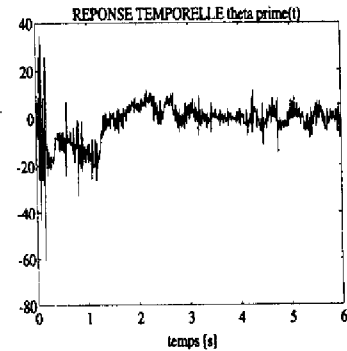


Figure 3.22 : $\dot{\theta}$.

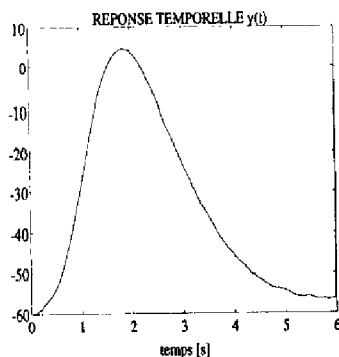


Figure 3.23 : y .

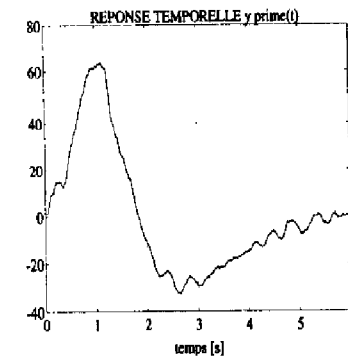


Figure 3.24 : \dot{y} .

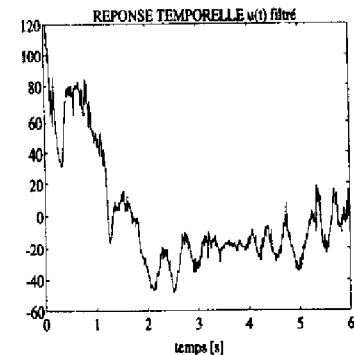


Figure 3.25 : u .

La réponse angulaire et le signal de référence angulaire, généré pendant le déplacement du chariot montrent que les oscillations de la réponse angulaire sont dues au suivi de la référence angulaire. L'erreur sur la position horizontale est d'environ 2cm. Le temps de réponse est d'environ 4s pour une condition initiale de 12° .

Des expériences comparatives ont été menées avec un régulateur à placement de pôles¹. Il est constitué d'un observateur d'état, un algorithme de placement de pôles et un module de compensation de perturbations. Les dynamiques en boucle fermée peuvent être choisies avec des pôles variant de -20 à -3 . L'objectif de ces expérimentations est d'évaluer la robustesse en stabilité des régulateurs. Celle-ci a été mesurée grossièrement par la détection de la condition initiale angulaire maximale (dans le cas des perturbations mécaniques) et par le niveau maximal d'atténuation des signaux (pour les perturbations électriques), qui autorisent toujours une stabilisation de la tige.

Concernant les perturbations mécaniques, en utilisant les tiges moyennes et petites (ce dernier cas étant le plus difficile à réguler, car le moment d'inertie est le plus faible), le régulateur flou est toujours capable de stabiliser le pendule inversé et est opérationnel pour des valeurs d'angle allant jusqu'à 10° dans le premier cas et 6° dans le second cas. La figure (3.26) montre le résultat de la régulation floue avec une petite tige et une petite masse; les discontinuités correspondent à des perturbations appliquées manuellement. Dans ce dernier cas, le régulateur à placement de pôles équilibre très difficilement la tige avec les pôles égaux à -20 et échoue avec les pôles -3 . En présence de perturbations mécaniques, une robustesse en stabilité légèrement meilleure peut être notée avec le régulateur flou, comparé au régulateur à placement de pôles. Toutefois des études plus poussées doivent être menées.

Des expériences ont été réalisées avec différents types de perturbations électriques; le tableau (3.4) résume les mesures effectuées. Les chiffres indiqués sont les pourcentages d'atténuation maximale possibles. Le régulateur flou est soit légèrement plus robuste, soit conduit à des résultats équivalents à ceux du RPP. La figure (3.27) correspond à 30% d'atténuation de θ et donc aussi $\dot{\theta}$, calculé numériquement.

Régulateur :	RF	RPP
θ and $\dot{\theta}$	30%	25%
U	15%	15%
Y	80%	80%
\dot{Y}	75%	70%

Tableau 3.4 : Perturbations électriques.

¹Régulateur algorithmique, fourni avec le pendule inversé et développé par la société *Amira*.

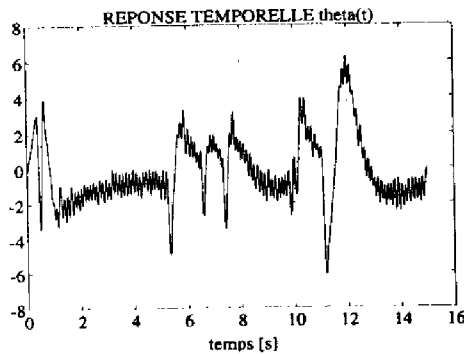


Figure 3.26 : Perturbations mécaniques, θ ($^{\circ}$).

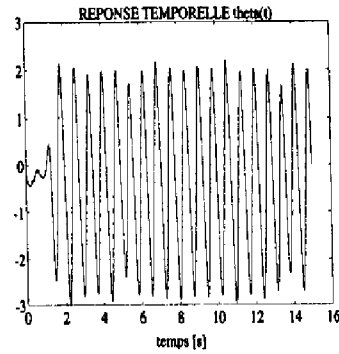


Figure 3.27 : Perturbations électriques sur θ et $\Delta\theta$, θ ($^{\circ}$).

3.2 Applications sur les fours

Considérons maintenant des problèmes de régulation de température avec un modèle simulé de four nominal. Le deuxième ordre possède un gain de 0.87, une constante de temps de 200s et une période d'échantillonnage de 10s -voir le paragraphe (1.4.2) du chapitre (1)-.

3.2.1 Réglage des RF utilisés

La structure de base des régulateurs flous est choisie de type *proportionnel-intégral*, avec inférence en ligne des variations de commande. L'action intégrale est utilisée afin d'obtenir des erreurs statiques nulles en régime permanent; en contrepartie, l'absence d'action dérivée génère un ordre sous-optimal du régulateur par rapport au système. Le choix d'une stratégie de régulation avec inférence en ligne est possible compte tenu de la lenteur des dynamiques des fours; elle permet d'éviter les erreurs d'approximation numérique introduites par le mécanisme de quantification. De plus, l'utilisation d'un régulateur flou avec inférence en ligne permettra de construire un régulateur ajustable, c.a.d. dont certains paramètres pourront être adaptés en ligne, de façon à optimiser les performances en boucle fermée, voir le chapitre (6).

Les ensembles flous, définissant les valeurs linguistiques des variables d'entrée, ont été choisis triangulaires, au nombre de 7 et disposés uniformément, avec un recouvrement de 1, sur les univers de discours correspondant. Concernant le module de connaissance linguistique, la construction de la base de règles est faite à partir d'une analyse qualitative. Cette démarche revient à construire la table de règles de *Macvicar-Whelan*, notée B_0 , pour un système linéaire. Les opérateurs du moteur d'inférence sont successivement: le *minimum* pour l'agrégation des prémisses des règles (*et*), le *maximum* pour l'agrégation des règles, la conjonction de *Mamdani* et la règle compositionnelle d'inférence *Sup.min*. La loi de régulation obtenue à ce stade,

avec un parcours complet des univers de discours des variables d'entrée du régulateur, est une loi de type PI, quasi-linéaire, saturée. Le régulateur flou choisi possède des ensembles flous sous forme de singletons pour la variable de sortie du régulateur. La raison d'un tel choix est une mise en place ultérieure d'une technique d'adaptation en ligne des conclusions de certaines règles.

Les bornes des univers de discours des variables entrée-sortie du régulateur sont fixées par analogie avec les facteurs proportionnel et intégral d'une loi de régulation PI analytique. L'équation (3.6) exprime la loi de régulation PI floue équivalente. Les paramètres à choisir par l'utilisateur, en fonction du système considéré, sont les facteurs $(K_p, T_i, \Delta U_{max})$, définissant une certaine dynamique en boucle fermée initiale. Les calibrages des univers de discours du régulateur flou sont déduits des formules (3.7) et (3.8), où T_e est la période d'échantillonnage.

$$\Delta U(k) = \frac{\Delta U_{max}}{\Delta e_{max}} * \Delta e(k) + \frac{\Delta U_{max}}{e_{max}} * e(k) \quad (3.6)$$

$$\Delta e_{max} = \frac{\Delta U_{max}}{K_p} \quad (3.7)$$

$$e_{max} = \frac{T_i}{T_e} \frac{\Delta U_{max}}{K_p} \quad (3.8)$$

La dynamique initiale en boucle fermée est choisie rapide et donc assez oscillatoire; les dépassements seront réduits grâce aux non-linéarités introduites dans une deuxième phase du réglage, voir le chapitre(6). Les formules utilisées pour guider le choix de la dynamique désirée sont fonction du type de consigne considérée. Les formules de *Ziegler et Nichols* conduisent à une dynamique oscillatoire pour un échelon. Dans le cas d'une rampe, il est souhaitable d'améliorer la précision du suivi de la référence. A ces fins, il est utile de baisser T_i et d'augmenter K_p dans des proportions qui conduisent à une dynamique boucle fermée satisfaisante, comme le montre l'expression de l'erreur de suivi de rampe (3.9) - E représente la pente de la rampe, et K_s le gain du système-. Le régulateur doit donc être plus rapide dans le cas des rampes et les formules de *Ziegler et Nichols*, modifiées selon les équations (3.10) et (3.11), sont utilisées (T_u est la période critique et K_u est le gain critique, obtenus par la méthode de pré-réglage du relais, sur un niveau intermédiaire dans la plage de consignes considérées). Les dynamiques PI considérées sont successivement $(K_p = 2.3, T_i = 191)$ dans le cas des poursuites de rampes et $(K_p = 1, T_i = 200)$ dans le cas des régulations autour d'échelons.

$$\varepsilon(\infty) = \frac{E T_i}{K_s K_p} \quad (3.9)$$

$$K_p = 0.45 \times K_u \times 2.3 \quad (3.10)$$

$$T_i = \frac{T_u \times 0.83}{1.05} \quad (3.11)$$

3.2.2 Analyse des résultats

Les résultats de régulations de type PI analytique et floue standard équivalente sont comparés. Il s'agit d'observer les effets des approximations numériques liées à la granularité floue, donc d'analyser les effets des interpolations locales non linéaires.

Une granularité floue assez grossière est définie afin de se localiser entre deux points caractéristiques, donc dans une zone de non-linéarités locales. Ainsi, les influences des interpolations pourront être observées. Ainsi, les bornes des univers de discours de l'erreur et de sa variation sont fixées à de grandes valeurs. La conséquence est que la zone de la table de règles parcourue pendant les simulations de régulations et de poursuites correspond aux labels (NP,ZE,PP); dans le cas des échelons, les ensembles flous PM et PG de Δe sont utilisés, compte tenu des initialisations opérées en début de régime transitoire. Le paramètre Δu_{max} , fixé à 120, permet de générer une loi de régulation de type PI non-saturée. En effet, dans le cas des échelons, la dynamique ($K_p = 1, T_i = 200$) conduit au calibrage ($e_{max} = 2400, \Delta e_{max} = 120$), et dans le cas des rampes, la dynamique ($K_p = 2.3, T_i = 191$) conduit au calibrage ($e_{max} = 996, \Delta e_{max} = 52$).

Les courbes (3.28) à (3.31) présentent respectivement: les deux lois de régulation pour les poursuites de rampes, les lignes de niveaux correspondantes -projections des lois de régulation $\Delta U(e, \Delta e)$ sur le plan de phase $e(\Delta e)$ -, les coupes selon les deux diagonales de la table de règles et les différences entre les lois PI analytique et floue standard.

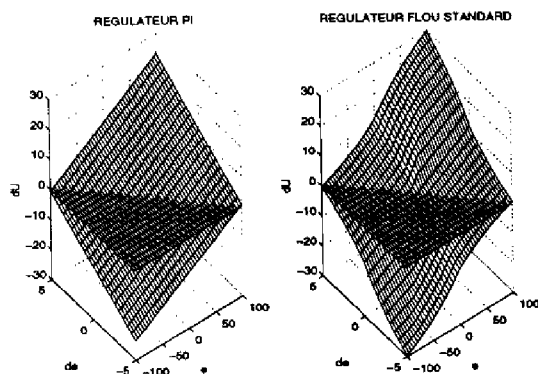


Figure 3.28 : $\Delta U(e, \Delta e)$ avec les régulateurs PI analytique et floue standard.

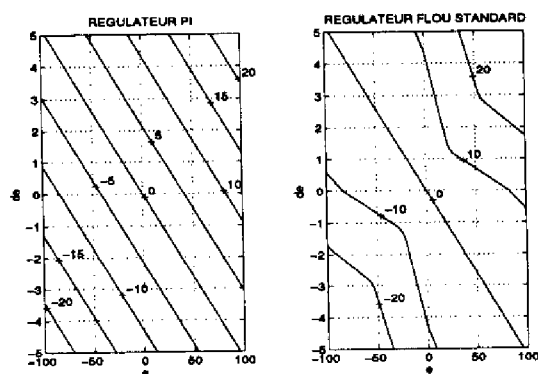


Figure 3.29 : Lignes de niveaux correspondantes.

Le seul point d'approximation exacte est l'origine du repère, qui correspond à des valeurs modales. Les erreurs d'approximation sont maximales lorsque l'erreur et sa variation sont de même signe (zones de divergence), l'amplitude de la variation de commande est alors renforcée. Dans les autres cas (zones de convergence), les valeurs absolues de ΔU ont tendance à être diminuées. Ces phénomènes locaux conduisent à la non-modification de la ligne de niveau zéro, car les erreurs d'approximations se compensent le long de la diagonale $(e, \Delta e) = (PP, NP) \Rightarrow (NP, PP)$, et à une accentuation de la pente de la loi de commande le long de l'autre diagonale.

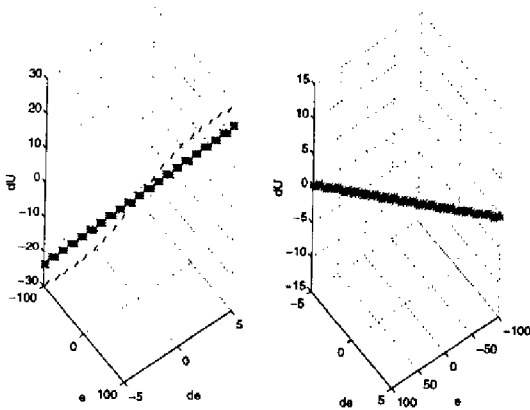


Figure 3.30 : Coupe des lois de régulation: ***: PI; - -: RF.

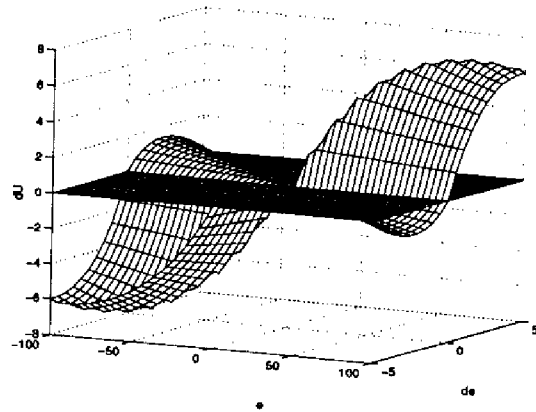


Figure 3.31 : Différence entre les lois floue standard et PI analytique.

Les courbes (3.32) et (3.33) présentent des simulations respectivement pour les poursuites de rampes et les régulations autour d'échelons.

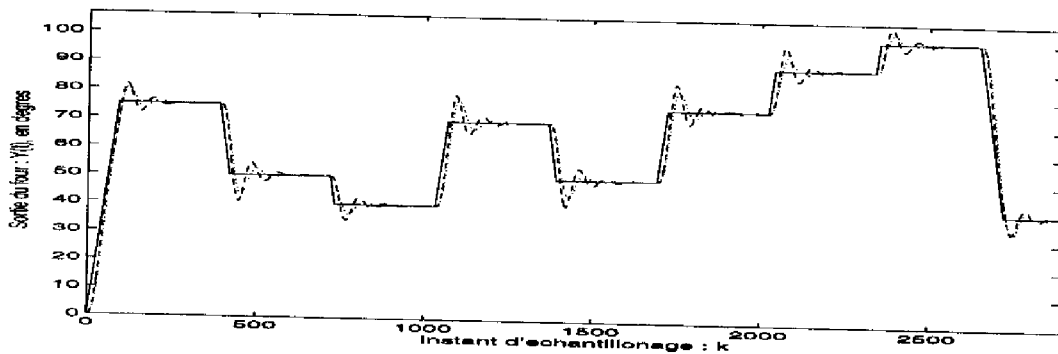


Figure 3.32 : Réponses comparatives du four à des rampes: ...: PI; - -: RF.

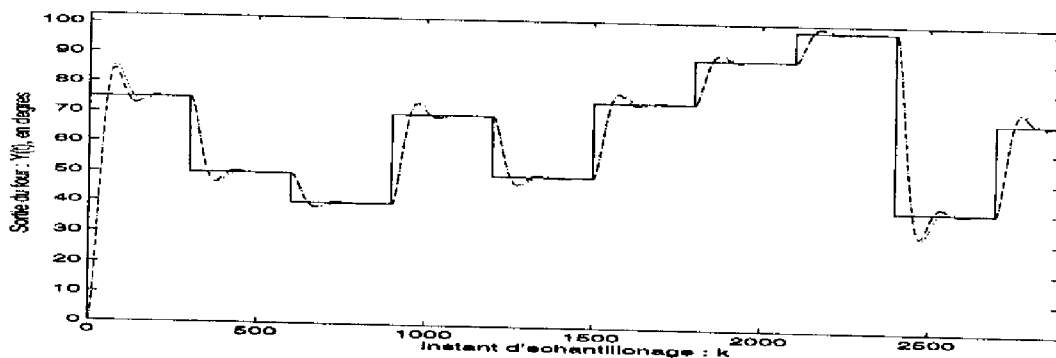


Figure 3.33 : Réponses comparatives du four à des échelons: ...: PI; - -: RF.

La figure (3.34) permet d'indiquer les parcours de la table de règles, pour les deux cas de simulation et de comprendre les conséquences sur les réponses temporelles.

Les indications sous forme de signes pour les différentes zones a, b,..., indiquent les différences entre la loi de commande floue et la loi PI linéaire. Dans la suite, est faite une analyse qualitative schématisée de l'influence des zones du plan de phase linguistique. Les conclusions fournies sont des tendances générales, dans la mesure où l'importance des différentes influences dépend de la position exacte dans le plan de phase linguistique et des effets combinés des zones adjacentes.

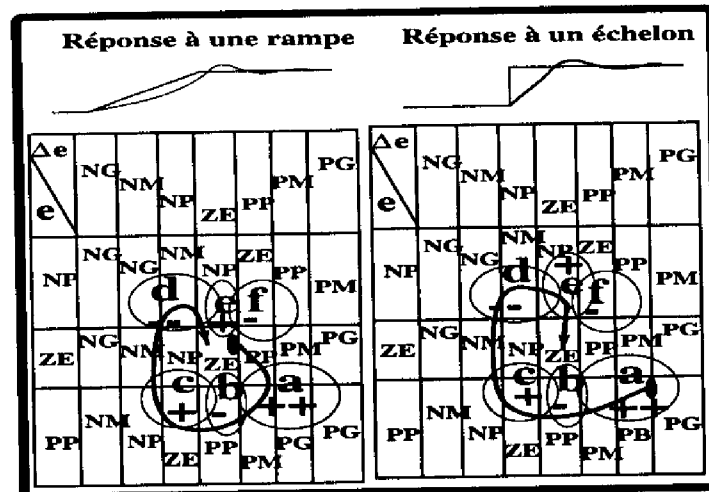


Figure 3.34 : Parcours de la table de règles dans le cas d'un échelon et d'une rampe.

La comparaison du RF par rapport au régulateur PI montre que, dans le cas des rampes, la zone a entraîne une anticipation plus rapide à la divergence en début de transitoire. La zone b correspond à une atténuation plus importante des commandes; ce qui explique qu'en fin de rampe, le suivi est équivalent au cas linéaire. La zone c liée à l'arrivée sur le plateau (portion de référence constante) est caractérisée par une augmentation plus importante des commandes, donc par des dépassements plus importants. Les zones d et f (e exerçant une action opposée) entraînent globalement une diminution plus importante des commandes, donc le premier dépassement en bas est plus important.

Les réponses données permettent de vérifier ces points. L'amélioration observée avec la loi floue standard par rapport à la loi PI analytique est un meilleur suivi de rampe, alors que le premier dépassement et le comportement oscillatoire sont plus accentués (pourcentage moyen de dépassement maximal: 12.7% pour le régulateur flou standard et 9.7% pour le régulateur PI; temps de stabilisation moyen: 45.7mn pour le régulateur flou standard et 26.8mn pour le régulateur PI).

Dans le cas des échelons, les zones a et b (actions opposées) ont globalement pour conséquence d'augmenter la commande, donc de produire une réponse plus rapide. Les zones d et c (actions opposées) provoquent de façon prédominante une diminution plus importante de u , atténuant le premier dépassement. Les effets combinés des zones

d et f (e exerçant une action opposée) génèrent un deuxième dépassement équivalent ou un peu plus bas.

Les réponses temporelles confirment ces faits. Le temps de montée est un peu meilleur avec le régulateur flou standard, l'amplitude du premier dépassement est légèrement atténuée, tandis que les oscillations suivantes sont un peu amplifiées (pourcentage moyen de dépassement maximal: 6.5% pour le régulateur flou standard et 7.2% pour le régulateur PI; temps montée à 100% moyen: 8.2mn pour le régulateur flou standard et 8.4mn pour le régulateur PI).

Le tableau ci-contre résume les effets locaux des interpolations non linéaires pour les échelons et les rampes; *P* signifie positif et *N* négatif.

	a	b	c	d	e	f
rampe	P	N	N	N	P	N
échelon	P	N	N	P	P	N

Globalement, les effets positifs sur la forme de la réponse temporelle sont la conséquence des zones de divergence, où les valeurs absolues des variations de commandes ont été amplifiées (zones a et d). Les effets négatifs correspondent aux actions de diminution des valeurs absolues de Δu rencontrées dans les zones de convergence (zones c et f). Les non-linéarités dues à l'inférence peuvent donc avoir des effets négatifs ou positifs; leur bonne utilisation passe par une maîtrise des opérateurs de l'inférence. Dans le cas où beaucoup de règles sont utilisées, ces effets sont très locaux et donc très peu influants. Par contre, le développement de la base de règles est délicat. Si peu de règles sont utilisées, le phénomène considéré est moins local et il est, alors, plus important de maîtriser les interpolations non linéaires.

Des améliorations sont nécessaires; les adaptations peuvent se faire par modification du type d'inférence ou par modification des règles de régulation. Elles seront apportées par l'introduction de non-linéarités en adaptant certaines règles, par une procédure d'auto-réglage dans le chapitre (6). Les actions d'adaptation prévues dans la base de règles, pour générer une loi de régulation floue plus fortement non linéaire, permettront de compenser les effets néfastes de l'interpolation grossière rencontrés dans les zones de convergence. Il s'agira de renforcer localement les actions d'anticipation sur les dépassements, de façon à mieux doser les commandes.

3.3 Conclusions et extensions

Conclusions

Les expériences menées sur différents modèles du pendule inversé ont permis de tester des structures de régulation floue de type PD, mises en oeuvre avec tables de décision.

Une méthodologie de réglage heuristique a été expérimentée. Bien que ne garantissant pas un réglage optimal, elle est facilement utilisable.

Une particularité intéressante de certaines lois de régulation floue est l'obtention de temps de réponse plus courts qu'avec des lois complètement linéaires. Nous pouvons relier cette propriété à la présence, dans la loi de régulation, de saturations et de non-linéarités locales dues à l'inférence.

Toutefois, des erreurs gênantes existent en régime permanent. Deux causes expliquent ce résultat: l'absence d'action intégrale et les imprécisions dues à l'utilisation de niveaux de quantification. Deux techniques permettent de diminuer ces imprécisions. La première consiste à utiliser plusieurs niveaux de réglage fins successifs, à mesure que le régime permanent approche. La mise en oeuvre nécessite plusieurs RF qui ne diffèrent que par leur matrice de quantification; une difficulté réside dans le choix des méthodes de commutation. La technique proposée est une interpolation linéaire en sortie d'une unique table de décision. La mise en place est très facile et la précision est nettement améliorée.

Nous avons également proposé une structure de régulation floue pour réguler le pendule inversé, à partir de la position angulaire initiale basse.

Des expériences temps-réel ont montré un léger apport des RF sur la robustesse en stabilité à des perturbations mécaniques et électriques, par comparaison avec un régulateur à placement de pôles.

Les expériences menées sur un modèle de four ont permis de tester des structures de régulation floue de type PI, mises en oeuvre avec inférence en ligne. La démarche de réglage, utilisant les règles d'équivalence avec les régulateurs conventionnels, a permis d'étudier les causes locales sous-jacentes des différences avec les lois de régulation linéaire. La nature des interpolations non linéaires dues à l'inférence a été mise en relation, dans les cas de consignes en rampes ou en échelons, avec certaines améliorations (régime transitoire ou dépassements dans certains cas).

Extensions

Quelques points abordés ou études commencées méritent une attention supplémentaire. Une étude comparative de robustesse en performances doit par exemple être menée sur le pendule inversé.

De plus, des améliorations peuvent être apportées au RF. Il faut fournir une méthodologie de réglage permettant d'introduire facilement des non-linéarités plus marquées et plus facilement maîtrisables que celles liées à l'inférence, par exemple à travers les règles de régulation floue.

Le chapitre (6) traitera de ces aspects; mais, auparavant, voyons quel peut être l'apport des réseaux de neurones à la commande....

Chapitre 4

Régulation neuronale

Les réseaux de neurones apparaissent potentiellement très intéressants pour des applications en régulation, pour leur caractère adaptatif et leur capacité à manipuler des données expérimentales, voir le paragraphe (1.2.1) du chapitre (1). Ainsi, la construction empirique de lois de régulation par *apprentissage* permet de générer implicitement des fonctions non linéaires, sans utiliser de modèles analytiques précis du système.

Nous avons opté pour une structure de régulation directe avec apprentissage spécialisé, qui nous semblait intéressante pour une première évaluation des techniques neuronales. Nous proposons et justifions une méthode de conception conduisant à une loi de régulation de type *régime glissant*. Nous proposons également des approches pour réduire les approximations dans les calculs des adaptations des poids du réseau neuronal. Le régulateur a été appliqué sur différents modèles du pendule inversé, puis sur le système réel.

Dans le paragraphe (4.1) de ce chapitre, sont fournis des éléments de présentation générale des outils neuronaux et des régulateurs neuronaux. Dans le paragraphe (4.2) est explicitée la méthodologie employée pour la conception du régulateur neuronal. Dans le paragraphe (4.3) sont décrits les résultats obtenus sur le pendule inversé, simulé ou réel. Finalement, conclusions et extensions sont fournies en (4.4).

4.1 Les réseaux de neurones: introduction

4.1.1 Dates dans l'étude des réseaux neuronaux artificiels

Cette partie permet d'indiquer les dates clés et les notions fondamentales du domaine neuromimétique. Les trois références [LIP,87], [DON,93] et [RUM,88] permettent de compléter ces informations.

A- La naissance des neurones artificiels

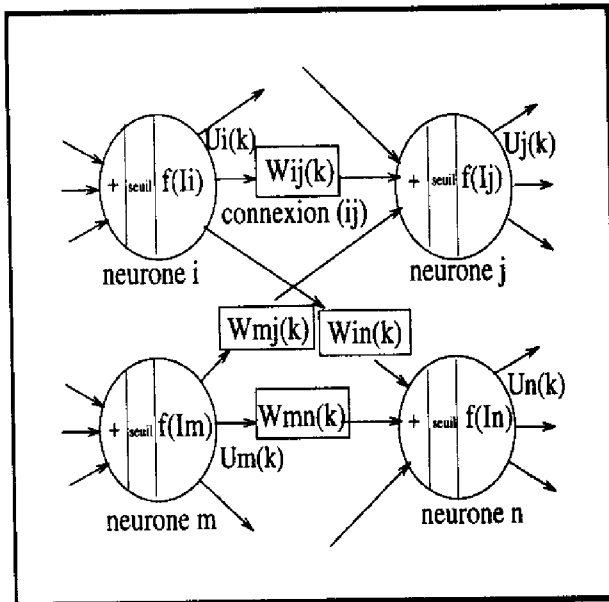


Figure 4.1 : Neurones artificiels.

Un neurone est une simple unité de calcul comprenant des fonctions de sommation, de seuillage, d'activation linéaires ou non linéaires (par exemple des droites saturées, des sigmoïdes, des gaussiennes....). Plusieurs processeurs élémentaires de ce genre sont reliés entre eux par des connexions de nature excitatrice ou inhibitrice, caractérisées par un poids synaptique, voir la figure (4.1). Le premier neurone de *McCulloch & Pitts* apparaît en 1943; il réalise des fonctions booléennes élémentaires.

La notion d'**apprentissage** émerge en 1949. L'idée est d'intégrer un comportement donné dans le réseau de neurones, en faisant évoluer ses poids synaptiques selon une technique d'adaptation. *Hebb* propose la règle d'apprentissage corrélacionnelle (4.1): un poids augmente d'autant plus que le neurone en amont excite le neurone en aval.

$$W_{ij}(k+1) = W_{ij}(k) + \alpha U_i(k) \times U_j(k) \quad (4.1)$$

Rosenblatt propose en 1957 la loi d'apprentissage *Delta* pour le Perceptron à une couche (les neurones ont une fonction d'activation relai et un seuil). Il est ainsi possible de classer des données, binaires ou continues, linéairement séparables. Les frontières

entre les classes sont des droites fonction du seuil d'un neurone et des poids appliqués aux signaux d'entrée.

En 1960, *Widrow & Hoff* proposent la loi d'adaptation *Delta* pour l'Adaline (où les fonctions d'activation de l'unique couche sont linéaires et les signaux continus), introduisant ainsi le principe de l'**apprentissage supervisé**: après l'application d'une entrée, la sortie désirée (t_{jp}) est spécifiée au réseau. La technique d'apprentissage, appelée règle des moindres carrés (*L.M.S. rule*), consiste à minimiser une erreur quadratique des sorties du réseau (indices j) sur un ensemble de p exemples, voir l'équation (4.2). La résolution est effectuée par une méthode de type gradient, voir la formule (4.3) (e_{ip} y représente l'entrée d'indice i). La procédure d'apprentissage peut être appliquée après chaque présentation d'un exemple avec une bonne convergence, si le taux d'apprentissage α est suffisamment petit et si les exemples sont présentés plusieurs fois.

$$E = \sum_p E_p = \sum_p \sum_j (u_{jp} - t_{jp})^2 \quad (4.2)$$

$$W_{ij}(k+1) = W_{ij}(k) - \alpha \sum_p \{ 2 e_{ip} (t_{pj} - u_{jp}) \} \quad (4.3)$$

B- L'émergence du connexionisme

Dans les années soixante, le *connexionisme* naît de la combinaison d'une partie de la cybernétique (étudiant des systèmes possédant des facultés d'auto-apprentissage) et de l'IA (dont l'étude concerne les systèmes symboliques). Dans le premier cas, les études se rapprochent de la neurophysiologie et dans le second cas, de la psychologie. Des structures neuronales hautement **parallèles** apparaissent alors (en anglais, il est question du P.D.P.: *Parallel Distributed Processing*), [RUM,88].

Grossberg introduit la notion d'**apprentissage non supervisé** en 1964: les sorties correspondant à un vecteur d'entrée sont inconnues. Un apprentissage compétitif est mené sur un réseau auto-organisateur, l'information est automatiquement extraite de ses entrées. Les neurones d'une couche sont organisés en groupe, avec des liaisons internes inhibitrices, de façon à représenter les caractéristiques des entrées. Les poids correspondants aux neurones les plus activés dans un groupe sont augmentés au détriment des autres; ils convergent alors vers la valeur d'une caractéristique. Des problèmes de stabilité se posent, se traduisant par l'altération d'information déjà apprise. Une version stabilisée a été développée, il s'agit du modèle ART (*Adaptive Resonance Theory*); un cycle de passages avant et arrière, à travers le réseau, pour un signal en entrée, permet de reconnaître une configuration déjà stockée ou d'en ajouter une nouvelle.

A la fin des années soixante, *Minsky & Papert* ont montré que le Perceptron mono-couche ne peut traiter que les problèmes linéairement séparables. Les recherches sur les réseaux de neurones ont alors été considérablement freinées, l'apprentissage supervisé étant impossible avec des couches intermédiaires.

C- Les évolutions récentes

Hopfield en 1982 propose un réseau, pour lequel est définie une fonction globale d'énergie; l'évolution des états du réseau correspond à une minimisation de cette fonction.

En 1986 apparait une méthode d'apprentissage pour un Perceptron multi-couches (réseau avec des connexions dirigées de ses entrées vers ses sorties), permettant de réaliser un apprentissage supervisé pour les neurones des couches cachées. Cette technique d'**apprentissage par rétropropagation** (ou règle Delta généralisée) est présentée en 1986 par *Rumelhart*, mais d'autres noms restent attachés à cette méthode: *Parker* (1982), *Werbos* (1974) et *Bryson & Ho* (1969). Un problème d'optimisation, dont le critère est une fonction des erreurs de sortie du réseau est résolu par une technique de type gradient. Une des conditions d'application est donc la dérivabilité des fonctions d'activation (par exemple sigmoïde ou à base radiale).

Cette nouvelle forme d'apprentissage va apporter aux Perceptrons multi-couches la propriété d'approximation universelle -voir les paragraphes (1.2.1) du chapitre (1) et (4.2.4) de ce chapitre-. De plus, des problèmes de classification plus complexes peuvent être résolus. En effet, considérant des fonctions d'activation de type relais avec seuil, la première couche du réseau sépare linéairement les entrées par des hyperplans. Les couches suivantes effectuent des traitements spécifiques sur les données (par exemple, les neurones de la seconde couche, en réalisant des *et* logiques intersectant les plans, génèrent des ensembles convexes; les neurones d'une troisième couche, en réalisant des *ou* logiques, réunissent ces régions pour définir des domaines quelconques). L'utilisation des fonctions sigmoïdes génèrent des domaines aux frontières un peu plus difficile à analyser, mais permettent de traiter des problèmes de classification plus hardus, [LIP,87].

Le choix des fonctions d'activation a une influence sur la nature des traitements effectués. Les fonctions sigmoïdes, couvrant entièrement l'espace d'entrée, conduisent à un traitement global des données. Les fonctions à base radiale permettent de traiter localement les données. L'intérêt de cette deuxième approche est un apprentissage plus rapide, la connaissance déjà accumulée pouvant être moins facilement détruite. Par contre des problèmes de dimensionalité peuvent se poser, [BER,94].

Les techniques d'**apprentissage faiblement supervisé** se sont développées dans les années 80. Il est question d'apprentissage par renforcement: le superviseur donne des indications qualitatives et les adaptations sont faites en utilisant des procédures d'essais/erreurs. Citons, par exemple: *Sutton & Barto* pour la méthode des différences temporelles permettant de réaliser de la prédiction, *Watkins*, *Glorennec* et *Berenji* pour le Q-learning.

Récemment sont apparus des réseaux dynamiques. Il s'agit de réseaux avec des neurones à retard (1986) ou des **réseaux récurrents**, qui sont une extension du réseau de *Hopfield* (1988). Ces derniers possèdent de nombreux bouclages internes; l'apprentissage s'effectue alors par convergence itérative vers un équilibre.

D- Classement des réseaux de neurones

Nous donnons ci-dessous, sans être exhaustive, quelques classes de problèmes pouvant être résolus par des réseaux de neurones:

Mémorisation: Cette fonction est réalisée de façon associative: l'information est stockée de manière distribuée dans le réseau. Des comportements peuvent être reproduits par association d'idées ou adressage par contenu. Un contenu mémoire peut être retrouvé à partir d'informations altérées. Mais globalement ces *mémoires auto-associatives* (réseau de *Hopfield*), ou *hétéro-associative* (réseau ART) possèdent un code correcteur d'erreur de faible qualité.

Classification: Certains réseaux permettent de séparer les vecteurs d'entrée en plusieurs classes pré-définies, comme par exemple le *perceptron*. D'autres réseaux réalisent la constitution de différents groupes d'après les propriétés des entrées (*clustering*). Par exemple, les *cartes auto-organisatrices* de *Kohonen* donnent une représentation de la distribution des entrées. Le réseau de *Grossberg* est un autre exemple; il différencie les entrées en fonction de leurs propriétés intrinsèques.

Identification: Certains réseaux, permettant de faire de l'identification, sont souvent utilisés en boîte noire: les deux phases identification de structure et estimation de paramètres sont combinées. Ainsi le Perceptron multi-couches, avec les procédures d'apprentissage par rétropropagation ou par renforcement, permet de jouer ce rôle. D'autre part, les réseaux récurrents permettent d'intégrer physiquement la dynamique des systèmes identifiés. Les régulateurs neuronaux sont souvent construits à partir des techniques d'identification.

4.1.2 Les réseaux utilisant la rétropropagation

Deux approches peuvent être envisagées: d'une part l'utilisation de réseaux pré-réglés hors ligne par **apprentissage généralisé** et, d'autre part, de réseaux adaptés en ligne, utilisant l'**apprentissage spécialisé**. Dans ce dernier cas, il est possible de définir les **structures de régulation directe** et les **structures de régulation indirecte**. La loi de régulation peut traduire la fonction dynamique inverse du système ou une simple loi d'asservissement, voir [HUN,92] et [NAR,90] pour plus de détails.

A- Modélisation de systèmes dynamiques et de l'inverse d'une fonction

Trois approches existent pour modéliser une dynamique par un réseau neuronal. La première consiste à utiliser un réseau de neurones statique (sans rebouclage) et à fournir en entrée du réseau tous les signaux historiques (les sorties et commandes précédentes), nécessaires pour traduire correctement la fonction dynamique E-S.

Une deuxième approche consiste à coder la dynamique dans un réseau récurrent. Grâce aux rebouclages, l'ordre du procédé peut être découvert implicitement lors de l'apprentissage.

Un certain nombre de conditions doivent être remplies pour pouvoir capturer la dynamique inverse d'un procédé. A tout instant, l'état du système doit être reconstituable à partir des p dernières sorties et des $(p - 1)$ dernières commandes (p est l'ordre du système). De plus, il doit exister une commande unique permettant d'atteindre une cible dans d instants (d : retard du système) et l'inverse du système doit être stable. Enfin, le réseau doit pouvoir approcher parfaitement la loi de commande dans la région de fonctionnement avec une faible vitesse d'adaptation de ses poids, [SAE,93]. Une solution idéale serait d'obtenir une loi de commande sous forme de la fonction dynamique inverse du système. Or c'est un problème difficile si l'ordre dépasse 1, [PSA,87]. L'apprentissage d'une simple loi d'asservissement s'avère alors souvent suffisante, comme en témoigne l'utilisation massive des algorithmes PID.

B- Réseaux pré-réglés utilisant l'apprentissage généralisé

Cette approche consiste à apprendre hors ligne une loi de régulation à partir d'une base d'exemples, issus du système à réguler. Dans ce cas, la fonction inverse du système est apprise -voir la figure (4.2)-.

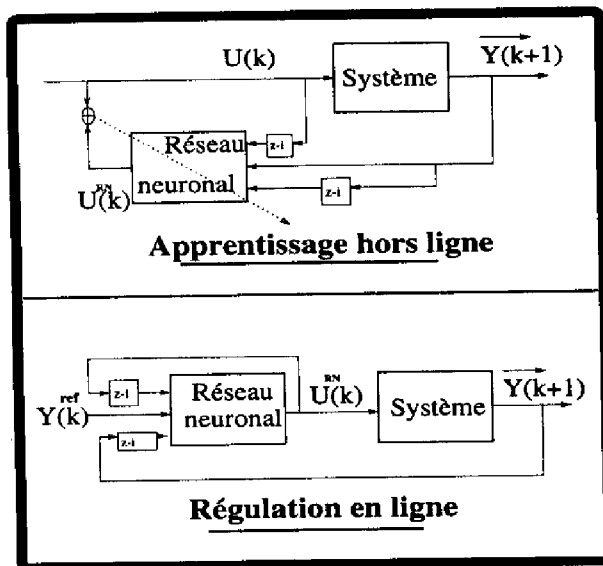


Figure 4.2 : Régulation neuronale avec apprentissage généralisé (\vec{Y} correspond à un vecteur de plusieurs sorties).

Il est question d'*apprentissage en regardant*. Dans un deuxième temps, le régulateur résultant est utilisé en ligne, voir la figure (4.2). Les limites de cette approche sont, tout d'abord, que le modèle du système doit être inversible. De plus, l'apprentissage n'est pas dirigé par les objectifs, et il n'est pas évident de choisir les valeurs de commande pertinentes pour l'apprentissage. Enfin, le régulateur utilisé (en ligne) est statique, l'adaptation étant opérée hors ligne.

C- Réseaux adaptés en ligne, en utilisant l'apprentissage spécialisé

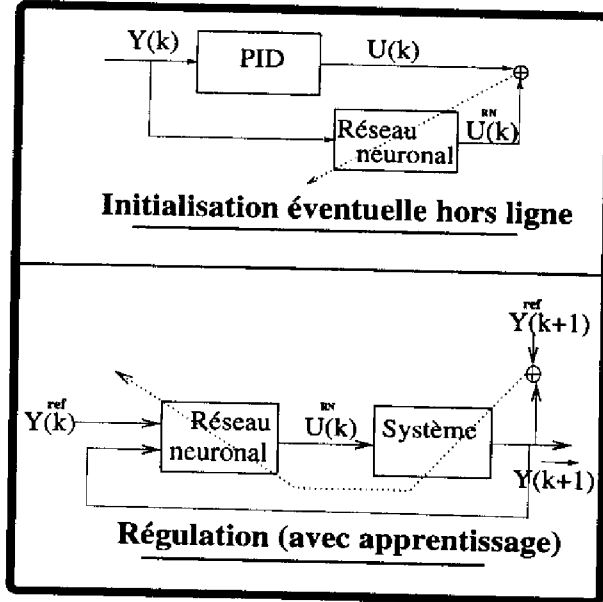


Figure 4.3 : Régulation neuronale avec apprentissage spécialisé.

Cette approche consiste à réajuster l'apprentissage en ligne, pendant la régulation, en fonction de l'erreur de sortie du système, voir la figure (4.3), [PSA,87] [SAE,91]. Ce dernier est assimilé à la dernière couche du réseau de neurones. Il est alors question d'*apprentissage en agissant*. La nature de la loi de régulation apprise est la fonction inverse du système, dans le cas où le type de réseau ou ses entrées le permettent et que les conditions générales d'inversibilité sont vérifiées. Dans les autres cas, elle est une loi d'asservissement.

Une connaissance initiale peut être intégrée dans le réseau, hors ligne, grâce à un régulateur existant (par exemple un opérateur humain ou un régulateur linéaire), voir la figure (4.3). L'adaptation en ligne permettra ensuite d'affiner la loi de régulation, en introduisant éventuellement des non-linéarités.

Un premier avantage de cette technique de régulation, par rapport à l'approche précédente, est son caractère adaptatif. La structure de commande utilisée équivaut à une version adaptative avec un modèle de référence (type MRAC), voir le paragraphe (1.1.1) du chapitre (1) et [AST,89]. De plus, l'apprentissage est dirigé par les objectifs: il est logique de choisir le domaine de variation de la sortie du système, en fonction des domaines de fonctionnement désirés. Un autre avantage de la méthode est qu'elle peut être appliquée à des systèmes non-inversibles. Toutefois, un point délicat est une connaissance minimale nécessaire sur les sensibilités des sorties du système par rapport à son entrée, traduites par le vecteur jacobien. La structure de régulation directe de la figure (4.3) n'utilise pas de modèle du système, le vecteur jacobien doit donc être déterminé hors ligne, pour pouvoir rétropropager l'erreur de sortie du système. La structure de régulation indirecte comprend un modèle neuronal direct du système, à travers lequel sont transmis les signaux d'erreur vers le réseau; le modèle permet alors une estimation en ligne du jacobien du processus.

4.2 Conception du régulateur neuronal proposé

4.2.1 Description du régulateur neuronal

A- Choix d'une structure de régulation neuronale

a- Présentation générale de la structure choisie: Les schémas synoptiques des régulateurs neuronaux considérés, pour stabiliser les modèles du pendule à deux ou quatre variables de sortie, sont présentés sur les figures (4.4) et (4.5). Une structure de régulation neuronale directe avec apprentissage spécialisé a été choisie. Le réseau apprend à fournir le signal de commande, de telle sorte que les sorties du pendule soient modifiées, en accord avec le critère utilisé par la procédure de rétropropagation. Les entrées du réseau de neurones ne comprenant pas l'historique nécessaire pour tenir compte de l'ordre du système (elles comprennent seulement les dernières sorties du système), le régulateur neuronal ne tend pas à reproduire la fonction dynamique inverse du système, mais apprend implicitement une loi d'asservissement de type PD non linéaire.

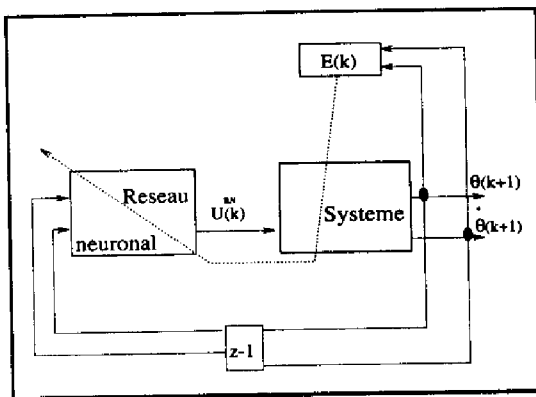


Figure 4.4 : Régulateur pour 2 sorties.

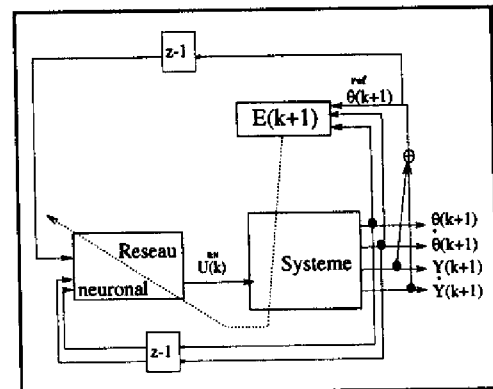


Figure 4.5 : Régulateur pour 4 sorties.

b- Justifications pour le choix effectué: La structure de régulation neuronale ne possède qu'un seul réseau neuronal, afin de ne pas traiter à la fois les problèmes d'identification de fonction et d'apprentissage d'une loi de régulation. De plus, nous n'avons pas opté pour la capture de la dynamique inverse globale du système. Pour pouvoir résoudre ce problème, il faudrait quatre signaux de commande. Si nous voulions inverser la dynamique liée à une sortie, par exemple θ -voir les équations récurrentes (1.8) du chapitre(1)- le signal $u(k-1)$ manquerait à l'entrée du réseau de la figure (4.5); l'information sur $\theta(k-1)$ est, elle, incluse dans $\dot{\theta}(k)$. Il faut alors un réseau avec plus d'entrées, donc plus de poids, ce qui rallonge les temps de calcul. De

toutes façons, il est impossible d'assurer la commandabilité de toutes les variables de sortie du pendule avec une seule commande. Nous nous sommes donc concentrée sur la construction expérimentale d'une loi d'asservissement.

B- Principe de fonctionnement du régulateur neuronal

L'aspect essentiel dans la conception du régulateur neuronal est l'apprentissage de la loi de régulation, par résolution d'un problème d'optimisation. Nous rappelons ci-dessous les équations d'adaptation des poids du réseau de neurones.

a- Technique d'apprentissage: La réalisation de l'apprentissage nécessite tout d'abord la définition d'une fonction d'erreur des sorties du système, qui constitue le critère d'un problème de minimisation. Les formules (4.4) et (4.5) indiquent respectivement le problème de minimisation posé et l'expression choisie pour le critère, où nb_s représente le nombre de sorties considérées. Un algorithme de type gradient est couramment utilisé pour résoudre ce problème. L'équation (4.6) exprime la solution du problème $-\vec{G}_r(k+1)$ est le vecteur gradient à l'instant $(k+1)$. Une méthode d'évaluation de ces gradients doit alors être choisie. Dans notre cas, il est possible de les exprimer analytiquement par un calcul des dérivées en chaîne, ce qui conduit à engendrer une rétropropagation des erreurs de sortie du système vers les entrées du réseau.

$$\underbrace{\text{MIN}}_{\vec{W}(k+1)} E(k+1) \quad (4.4)$$

$$\begin{cases} E(k+1) &= \frac{1}{2} \{ n_\theta [\theta(k+1) - \theta_{ref}(k+1)] + n_{\dot{\theta}} \dot{\theta}(k+1) \}^2 \\ \theta_{ref}(k+1) &= \begin{cases} - \{ n_y y(k+1) + n_{\dot{y}} \dot{y}(k+1) \} & \text{pour } nb_s = 4 \\ 0 & \text{pour } nb_s = 2 \end{cases} \end{cases} \quad (4.5)$$

$$\begin{cases} \vec{W}(k+1) &= \vec{W}(k) + \Delta \vec{W}(k+1) = \vec{W}(k) - \alpha \vec{G}_r(k+1) \\ \forall ij &: \Delta w_{ij}(k+1) = -\alpha G_{rij}(k+1) = -\alpha \frac{\partial E(k+1)}{\partial w_{ij}(k)} \end{cases} \quad (4.6)$$

b- Les équations d'apprentissage: La formule d'adaptation pour la couche cachée est la même que pour la rétropropagation classique, utilisant une fonction d'erreur définie directement à partir des sorties du réseau de neurones, voir la formule (4.7) et la figure (4.6) pour comprendre les notations utilisées. Les grandeurs ξ sont les termes d'erreur qui sont rétropropagés d'une couche à la précédente; le parcours de l'indice s correspond respectivement aux signaux θ , $\dot{\theta}$, y et \dot{y} . Le facteur α est le taux d'apprentissage et le terme γ est un terme de *moment*, qui permet d'allonger le pas de recherche sans engendrer de phénomènes oscillatoires (grâce à l'utilisation des

informations sur les actions menées précédemment).

Pour les poids de la couche de sortie du réseau, il existe des différences avec la rétropropagation classique dues à la présence du système avant les signaux utilisés par la fonction d'erreur, voir la formule (4.8). La différence avec la formule de la rétropropagation classique réside dans l'utilisation du jacobien du système (de composantes $J_s(k+1)$, $s = 1$ à 2 ou 4). La détermination des fonctions de sensibilité des sorties du pendule, par rapport à son signal d'entrée, est donc un aspect propre à cette technique de régulation neuronale.

$$\begin{cases} \Delta w_{hi}(k+1) = -\alpha \xi_i^{[l_s-1]}(k+1) u_h^{[l_s-2]}(k) + \gamma \Delta w_{hi}(k) \\ \xi_i^{[l_s-1]}(k+1) = \xi_j^{[l_s]}(k+1) w_{ij}(k) f'[I_i^{[l_s-1]}(k)] \end{cases} \quad (4.7)$$

$$\begin{cases} \Delta w_{ij}(k+1) = -\alpha \xi_j^{[l_s]}(k+1) u_i^{[l_s-1]}(k) + \gamma \Delta w_{ij}(k) \\ \xi_j^{[l_s]}(k+1) = \left[\sum_{s=1}^4 \xi_s^{[l_s+1]}(k+1) J_s(k+1) \right] f'[I_j^{[l_s]}(k)] \\ \xi_s^{[l_s+1]}(k+1) = \frac{\partial E(k+1)}{\partial y_s(k+1)} \quad \text{pour } s \in [1, 4] \end{cases} \quad (4.8)$$

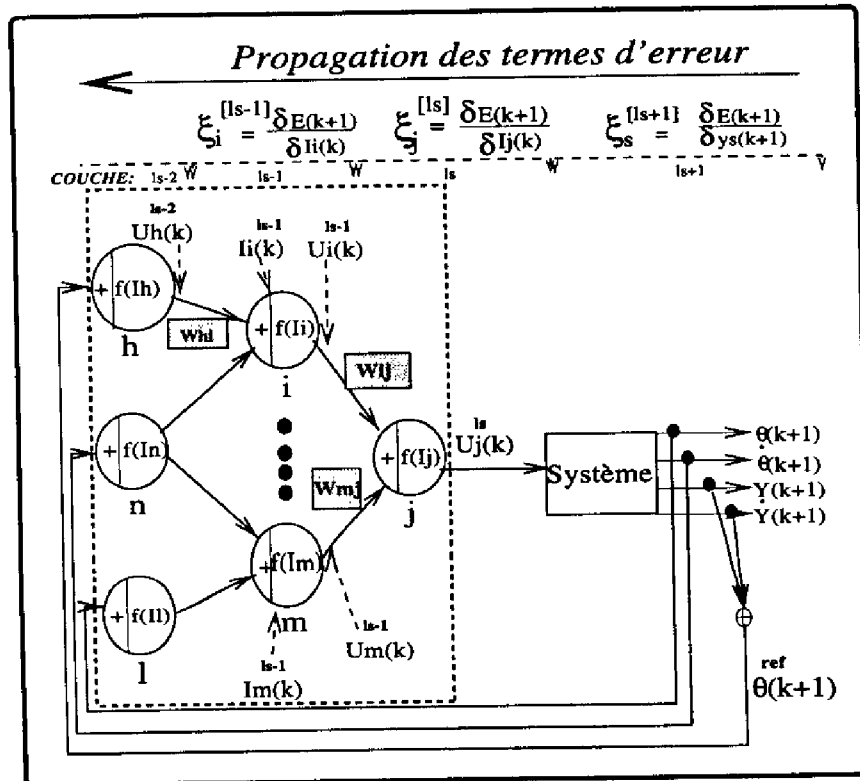


Figure 4.6 : Notations utilisées pour la procédure de rétropropagation.

c- **Eléments de démonstration des équations d'apprentissage:** La figure (4.7) permet de comprendre l'établissement des formules d'adaptation des poids du réseau neuronal pour notre application. Les équations d'adaptation généralisées de type Narendra résultent du calcul des composantes du gradient, en utilisant des dérivées en chaîne, appliquées aux fonctions E-S du système et du régulateur.

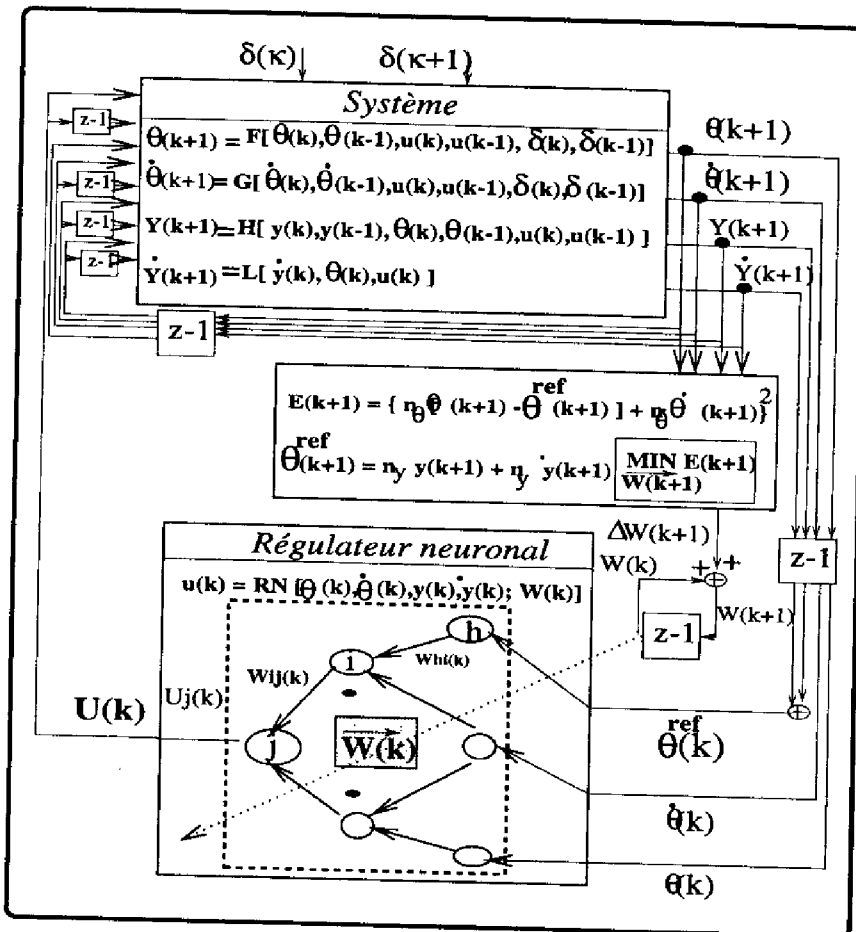


Figure 4.7 : Boucle fermée avec le régulateur neuronal.

Dans un premier temps, les sorties du système sont intégrées dans l'expression du gradient, voir l'équation (4.9). Ensuite, les équations récurrentes du système permettent de propager le calcul des termes d'erreur vers le réseau neuronal, voir l'équation (4.10) (où n_s vaut 2 pour $s = 3$ (sortie y) et 1 pour $s = 4$ (sortie \dot{y})). L'étape suivante consiste à faire intervenir les signaux dont dépend la sortie du réseau neuronal, voir l'équation (4.11). Dans cette formule le dernier terme en gras correspond au calcul de la rétropropagation classique à travers le seul réseau de neurones.

$$\forall ij : G_{rij}(k+1) = \sum_{s=1}^4 \frac{\partial E(k+1)}{\partial y_s(k+1)} \frac{\partial y_s(k+1)}{\partial w_{ij}(k)} \quad (4.9)$$

$s \in \{1, 2\} : (\theta, \dot{\theta})$

$$\frac{\partial y_s(k+1)}{\partial w_{ij}(k)} = \sum_{x=1}^2 \frac{\partial y_s(k+1)}{\partial y_s(k+1-x)} \frac{\partial y_s(k+1-x)}{\partial w_{ij}(k)} + \sum_{x=1}^2 \frac{\partial y_s(k+1)}{\partial u(k+1-x)} \frac{\partial u(k+1-x)}{\partial w_{ij}(k)}$$

$s \in \{3, 4\} : (y, \dot{y})$

$$\begin{aligned} \frac{\partial y_s(k+1)}{\partial w_{ij}(k)} &= \sum_{x=1}^{n_s} \frac{\partial y_s(k+1)}{\partial y_s(k+1-x)} \frac{\partial y_s(k+1-x)}{\partial w_{ij}(k)} + \sum_{x=1}^{n_s} \frac{\partial y_s(k+1)}{\partial y_1(k+1-x)} \frac{\partial y_1(k+1-x)}{\partial w_{ij}(k)} \\ &\dots + \sum_{x=1}^{n_s} \frac{\partial y_s(k+1)}{\partial u(k+1-x)} \frac{\partial u(k+1-x)}{\partial w_{ij}(k)} \end{aligned} \quad (4.10)$$

$$\frac{\partial u(k)}{\partial w_{ij}(k)} = \sum_{s=1}^4 \frac{\partial u(k)}{\partial y_s(k)} \frac{\partial y_s(k)}{\partial w_{ij}(k)} + \frac{\partial RN(k)}{\partial \mathbf{w}_{ij}(k)} \quad (4.11)$$

Avant de poursuivre la décomposition du calcul à travers le réseau, des expressions simplifiées des formules d'adaptation peuvent être fournies. Il s'agit des équations de type *Jordan* donnée en (4.12). Dans [REN,93] de telles approximations sont justifiées. La dynamique du système est ignorée compte tenu de la fréquence d'adaptation des poids à chaque instant d'échantillonnage. De plus, seule la dépendance de la sortie du réseau à ses paramètres est considérée. Les récurrences des équations de type *Narendra* étant éliminées, des formules beaucoup plus faciles et rapides à calculer sont obtenues. Le reste des développements mathématiques concernent maintenant le dernier terme de l'équation (4.12).

$$G_r(k+1) = \left\{ \sum_{s=1}^4 \frac{\partial E(k+1)}{\partial y_s(k+1)} \frac{\partial y_s(k+1)}{\partial u(k)} \right\} \frac{\partial RN(k)}{\partial \mathbf{w}_{ij}(k)} \quad (4.12)$$

Revenons maintenant sur les équations de rétropropagation à travers le réseau de neurones, données dans le paragraphe précédent en (4.7) et (4.8). Dans le cas de la couche de sortie, la décomposition de la formule (4.13) permet d'écrire l'équation d'adaptation de la dernière couche. Les dérivations en chaîne de la formule (4.14) permettent d'établir les équations d'adaptation pour la couche cachée.

$$\frac{\partial RN(k)}{\partial w_{ij}(k)} = \frac{\partial u_j^{[l_s]}(k)}{\partial I_j^{[l_s]}(k)} \frac{\partial I_j^{[l_s]}(k)}{\partial w_{ij}^{[l_s]}(k)} = f'[I_j^{[l_s]}(k)] u_i^{[l_s-1]}(k) \quad (4.13)$$

$$\begin{aligned} \frac{\partial RN(k)}{\partial w_{hi}(k)} &= \frac{\partial u_j^{[l_s]}(k)}{\partial I_j^{[l_s]}(k)} \frac{\partial I_j^{[l_s]}(k)}{\partial u_i^{[l_s-1]}(k)} \frac{\partial u_i^{[l_s-1]}(k)}{\partial I_i^{[l_s-1]}(k)} \frac{\partial I_i^{[l_s-1]}(k)}{\partial w_{hi}(k)} \\ &= f'[I_j^{[l_s]}(k)] w_{ij}(k) f'[I_i^{[l_s-1]}(k)] u_h^{[l_s-2]}(k) \end{aligned} \quad (4.14)$$

4.2.2 Loi de régulation neuronale proposée: régime glissant

Dans la suite, nous proposons une démarche pour fixer la trajectoire suivie dans le plan de phase, durant les régulations. Nous mettons en évidence le lien entre le critère d'apprentissage proposé et une loi de commande de type *régime glissant*.

A- Forme de la trajectoire dans le plan de phase

Les évolutions dans le plan de phase dépendent de la position cible (c'est-à-dire du minimum global du problème de minimisation) et des difficultés de convergence vers l'objectif fixé (c'est-à-dire, principalement, de l'existence éventuelle de minima locaux).

Les directions de convergence dans le plan de phase, pour chaque ajustement de poids, sont liées aux directions du gradient. La formule (4.15) donne l'expression générale du gradient pour le problème d'optimisation (4.4), dont la solution est donnée en (4.6); les formules (4.7) et (4.8) permettent de préciser les termes d'erreur intervenant dans l'équation (4.15). Les directions du gradient dépendent de la surface d'erreur, donc du critère choisi, qui définit finalement la stratégie de régulation. En effet, les poids sont ajustés de manière à réduire la distance entre la sortie et la cible; la cible peut être définie comme un modèle de référence.

Nous étudions dans la suite deux critères, tels que l'objectif final est identique, mais, les façons de l'atteindre sont différentes. Nous détectons, dans chaque cas, si la trajectoire de convergence est clairement définie et si des minima locaux peuvent interrompre le parcours dans le plan de phase.

$$\vec{G}_r(k+1) = \begin{pmatrix} \vec{G}_r^{[ls]}(k+1) \\ \hline \vec{G}_r^{[ls-1]}(k+1) \end{pmatrix} = \begin{pmatrix} \xi_j^{[ls]}(k+1) & u_i^{[ls-1]}(k) \\ \vdots & \vdots \\ \xi_i^{[ls-1]}(k+1) & u_h^{[ls-2]}(k) \\ \vdots & \vdots \end{pmatrix} \quad (4.15)$$

Les critères considérés étant positifs, le minimum global correspond à l'annulation du critère, voir l'équation (4.16). Le risque de présence de minima locaux peut être mis en évidence grâce aux conditions de stationarité (4.17). Les points solutions de cette relation sont, en toute généralité, des points stationnaires, c'est-à-dire des maxima, minima locaux ou globaux ou des points d'inflexion à dérivée nulle.

$$E(k+1) = 0 \quad (4.16)$$

$$\vec{G}_r(k+1) = \vec{0} \quad (4.17)$$

B- Stratégie classique

Le critère d'apprentissage classique des moindres carrés est donné en (4.18).

$$E^c(k+1) = \frac{1}{2} \sum_{s=1}^4 n_{y_s} y_s(k+1)^2 \quad (4.18)$$

Le minimum global est le point défini par des coordonnées nulles du vecteur de sortie du pendule. Cette information n'indique pas comment se fait la convergence dans le plan de phase vers l'état but. Avec ce critère classique, les variables de sortie sont forcées de converger vers le point cible, en utilisant dans le plan de phase le chemin réalisant au mieux le compromis, à chaque pas d'échantillonnage, entre les minimisations de $\theta(k+1)$, $\dot{\theta}(k+1)$, $y(k+1)$ et $\dot{y}(k+1)$.

Les conditions d'optimalité du premier ordre indiquent que les extréma sont caractérisés par: l'annulation des sorties du système, ou la relation (4.19), ou des configurations très spéciales des poids des couches cachées et de sortie du réseau (valeurs très grandes ou nulles). La relation (4.19) caractérise un extrémum qui pourrait être un minimum local.

$$\sum_{s=1}^4 n_s y_s(k+1) J_s(k+1) = 0 \quad (4.19)$$

Quelque soit la nature des extrema mis en évidence, le problème est que la trajectoire dans le plan de phase n'est pas connue à l'avance par le concepteur.

C- Critère proposé

Pour le critère proposé dans ce travail et formulé en (4.5), les minima globaux sont donnés en (4.20), [DEG,95b].

$$\begin{cases} \dot{\theta}(k+1) &= -\frac{n_{\theta}}{n_{\dot{\theta}}} [\theta(k+1) - \theta_{ref}(k+1)] \\ \theta_{ref}(k+1) &= \begin{cases} -\{n_y y(k+1) + n_{\dot{y}} \dot{y}(k+1)\} & \text{pour } s = 4 \\ 0 & \text{pour } s = 2 \end{cases} \end{cases} \quad (4.20)$$

Dans le cas de la régulation de θ seulement ($s = 2$), la procédure d'apprentissage tend à forcer les poids du réseau de neurones à converger vers un état, permettant au système bouclé de progresser le long d'une droite de glissement. Le sens de progression sur la droite correspond à la convergence vers l'état but $(0, 0)$, et non à une évolution vers les bornes infinies de la seconde bissectrice. Il est en effet mathématiquement impossible d'avoir à la fois un angle positif qui augmente et une vitesse angulaire négative, voir la figure (4.8). Le choix des deux facteurs n_{θ} et $n_{\dot{\theta}}$ permet d'imposer le temps de réponse du système régulé.

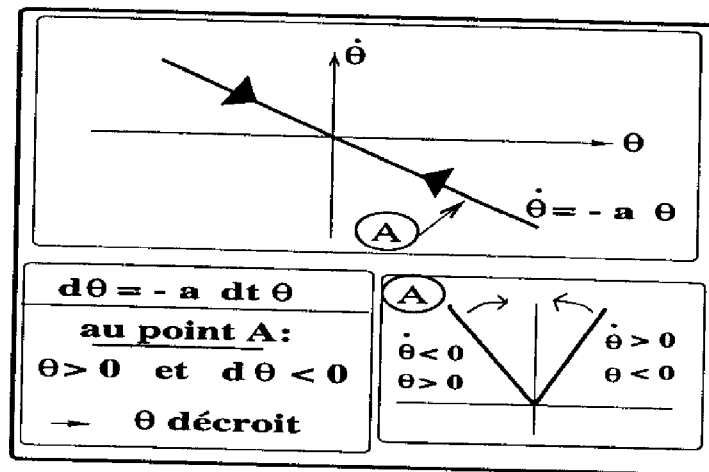


Figure 4.8 : Stratégie de commande de type régime glissant.

Dans le cas de la régulation de θ et y ($s = 4$), la stratégie de régulation est similaire à celle du régulateur flou, voir le paragraphe (3.1.4) du chapitre (3). L'objectif prioritaire est la régulation de θ et l'objectif secondaire est l'annulation de y . Pour cela, une référence angulaire est choisie à chaque instant d'échantillonnage, en fonction de la position et de la vitesse du chariot. La stratégie de régulation consiste à faire converger linéairement l'état du système vers le point $[\theta(k+1) = \theta_{ref}(k+1), \dot{\theta}(k+1) = 0]$ à chaque instant d'échantillonnage. Tant que l'angle de référence est non nul, des commutations sont opérées entre des droites de glissement de pente $\frac{n_\theta}{n_{\dot{\theta}}}$.

Les conditions d'optimalité du premier ordre montrent que les extréma du problème sont caractérisés par: des configurations très spéciales des poids des couches cachées et de sortie du réseau (valeurs très grandes ou nulles), ou l'appartenance aux droites de glissement (4.20), ou un couple $[\theta(k+1), \dot{\theta}(k+1)]$ nul avec une évolution du chariot le long de la droite de glissement $n_y y(k+1) = n_{\dot{y}} \dot{y}(k+1)$. Aucun minimum local n'existe alors et la convergence vers l'origine du plan de phase est garantie, si l'apprentissage est suffisamment long.

Avec le critère (4.5), la trajectoire suivie dans le plan de phase peut donc être pré-définie par le concepteur. *En résumé, l'intérêt des critères choisis réside principalement dans la possibilité de fixer explicitement les performances dynamiques attendues en boucle fermée.*

4.2.3 Propositions pour des calculs optimaux des directions du gradient

Le problème principal rencontré, lors de la mise en oeuvre de l'apprentissage, réside dans le calcul de la direction de plus grande pente (donc du gradient). L'adaptation

des poids du réseau de neurones se fait, en pratique, rarement selon cette direction optimale. La convergence vers le minimum est alors faite suivant une trajectoire approchée. Les raisons de ces imprécisions sont principalement la *fréquence d'adaptation*¹ des poids du réseau et les approximations faites sur le jacobien du système régulé.

A- Choix de la fréquence d'adaptation des poids

Un compromis préside le choix de la fréquence d'adaptation des poids du réseau. D'une part, il s'agit de se rapprocher de la détermination du vrai gradient, qui implique d'adapter après avoir appliqué plusieurs exemples. D'autre part, il faut agir le plus souvent possible à cause du fonctionnement en ligne du système. Une première approximation dans l'adaptation des poids est faite quand le gradient est calculé à chaque instant d'échantillonnage, auquel cas il est nécessaire d'utiliser un faible taux d'apprentissage. *Nous avons, dans un premier temps, adopté cette solution.*

Une autre idée consiste à adapter tous les trois instants d'échantillonnage, afin de tenir compte de la dynamique du système. La considération des influences des signaux E-S du système à l'ordre deux, permet aux commandes de se rapprocher de la référence en deux pas, voir la figure (4.9).

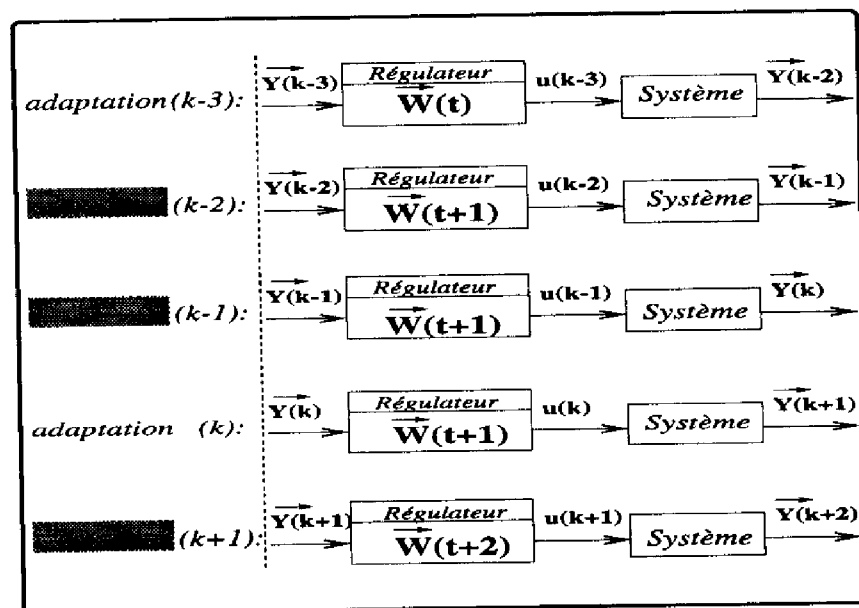


Figure 4.9 : Adaptation des poids pour tenir compte de l'ordre du système.

Les formules d'adaptation proposées pour cette démarche sont plus complexes que celles de type *Jordan*, tout en correspondant à une simplification des termes récurrents

¹Définition pour simplicité: nombre d'exemples appliqués avant de mener une adaptation.

inverses -de la forme $\frac{\partial y_s(k-i)}{\partial w_{ij}(k)}$ - des équations de type *Narendra*, voir la formule (4.10). Les équations (4.21) à (4.24) présentent respectivement: le critère, le gradient et les dérivées en chaîne pour s égal à 1 et 2 (pour s égal à 3 ou 4, le principe est le même). Le paragraphe suivant apportera une réponse à la manière d'évaluer les dérivées en chaîne introduites.

$$\left\{ \begin{array}{l} E_p(k+1) = \frac{1}{2} \sum_{p=0}^2 \{ n_\theta [\theta(k+1-p) - \theta_{ref}(k+1-p)] + n_\dot{\theta} \dot{\theta}(k+1-p) \}^2 \\ \theta_{ref}(k+1-p) = - \{ n_y y(k+1-p) + n_{\dot{y}} \dot{y}(k+1-p) \} \end{array} \right. \quad (4.21)$$

$$G_{rij}(k+1) = \sum_{p=0}^2 \left\{ \sum_{s=1}^4 \frac{\partial E(k+1-p)}{\partial y_s(k+1-p)} \frac{\partial y_s(k+1-p)}{\partial w_{ij}(t+1)} \right\} \quad (4.22)$$

$s \in \{1, 2\} : (\theta \text{ et } \dot{\theta})$

$$\left\{ \begin{array}{l} \frac{\partial y_s(k+1)}{\partial w_{ij}(t+1)} = \sum_{x=1}^2 \frac{\partial y_s(k+1)}{\partial y_s(k+1-x)} \frac{\partial y_s(k+1-x)}{\partial w_{ij}(t+1)} + \sum_{x=1}^2 \frac{\partial y_s(k+1)}{\partial u(k+1-x)} \frac{\partial u(k+1-x)}{\partial w_{ij}(t+1)} \\ \frac{\partial y_s(k)}{\partial w_{ij}(t+1)} = \frac{\partial y_s(k)}{\partial y_s(k-1)} \frac{\partial y_s(k-1)}{\partial w_{ij}(t+1)} + \sum_{x=1}^2 \frac{\partial y_s(k)}{\partial u(k-x)} \frac{\partial u(k-x)}{\partial w_{ij}(t+1)} \\ \frac{\partial y_s(k-1)}{\partial w_{ij}(t+1)} = \frac{\partial y_s(k-1)}{\partial u(k-2)} \frac{\partial u(k-2)}{\partial w_{ij}(t+1)} \end{array} \right. \quad (4.23)$$

$x \in \{0, 2\} :$

$$\frac{\partial u(k-x)}{\partial w_{ij}(t+1)} = \sum_{s=1}^4 \frac{\partial u(k-x)}{\partial y_s(k-x)} \frac{\partial y_s(k-x)}{\partial w_{ij}(t+1)} + \frac{\partial \text{RN}(k-x)}{\partial \mathbf{w}_{ij}(t+1)} \quad (4.24)$$

B- Evaluation du jacobien

Dans ce paragraphe, nous décrivons plusieurs méthodes pour évaluer le jacobien du système. Nous analysons également l'effet de l'approximation utilisée sur la stratégie de régulation.

Deux catégories de méthodes peuvent être distinguées, selon que les calculs se font hors ligne ou en ligne.

Parmi le premier type d'approche, il est possible de citer les calculs analytiques sur un modèle du pendule. Le problème n'est pas résoluble avec les équations différentielles non linéaires, mais il possède des solutions avec les équations linéarisées données dans le paragraphe (1.4.1) du chapitre (1). Par exemple, la composante du jacobien liée à θ est donnée en (4.25). Pour pouvoir mener à bien ce calcul, les dérivées première et tierce de θ sont nécessaires; le profil du signal $u(t)$ doit donc être connu à l'avance, pour pouvoir intégrer les équations différentielles linéaires.

$$J_\theta = \frac{B \dot{\theta}}{\ddot{\theta} - A \dot{\theta}} \quad (4.25)$$

D'autres méthodes sont souhaitables, compte tenu des limitations de l'approche analytique. Des évaluations numériques, selon l'équation (4.26), sont envisageables avec un modèle non linéaire (différentiel ou neuronal). Ainsi, les tables numériques à remplir ont pour entrées les signaux historiques et la commande courante, dans le cas d'un modèle de type ARMAX non linéaire, et les dérivées première et seconde des positions angulaires et du chariot, dans le cas d'équations différentielles.

$$\frac{\partial y_s(k+1)}{\partial u(k)} \approx \frac{\tilde{y}_s(k+1) - y_s(k+1)}{\tilde{u}(k) - u(k)} \quad (4.26)$$

Des méthodes moins précises peuvent également permettre d'établir les valeurs approximatives du jacobien du pendule. Ainsi, en menant un raisonnement approché avec un formalisme flou, il sera possible d'inférer les valeurs numériques des fonctions de sensibilité. Dans le domaine angulaire de fonctionnement linéaire (petits angles), les règles sont, par exemple, de la forme:

Si $u(k-1)$ est PP et $u(k)$ est PP et $\theta(k-1)$ est PG et $\theta(k)$ est PM (donc $\theta(k+1)$ est PP) et $\delta u(k)$ est PP Alors $\tilde{\theta}(k+1)$ est ZE (donc $J_\theta(k+1)$ est NP).

Les calibrages des univers de discours peuvent se faire en considérant des valeurs maximales physiques des signaux (90°C pour θ , 120N pour u , 1N pour δu ...), et, pour $J_{y_s \max}$, en effectuant quelques mesures numériques dans les configurations qui mènent aux fonctions de sensibilité maximales. Un inconvénient de cette approche est la longueur de l'étape d'établissement des règles.

Des évaluations qualitatives simplifiées peuvent alors être utilisées en approximant les jacobiens par leur signe. Dans le cas d'une position angulaire comprise dans $[-90^\circ, 90^\circ]$, **en phase de convergence vers les positions but** (situation atteinte lorsque le régulateur fonctionne), les signes des composants du jacobien sont négatifs pour θ et $\dot{\theta}$ et positifs pour y et \dot{y} . Les règles qui permettent de déduire ces faits sont du type: $\forall H(k)$ *Si $\delta u(k)$ est P Alors $\delta \theta(k+1)$ est N (donc $J_\theta(k+1)$ est N) -H(k) caractérise l'historique à l'instant (k), tel que les vitesses soient convergente-.*

Dans une deuxième approche, le jacobien peut être évalué numériquement en ligne entre deux instants d'échantillonnage, selon la formule (4.27). Toutefois la validité de cette approximation peut être remise en cause pour de grande valeurs de $\delta u(k)$. D'autre part, d'un instant d'échantillonnage au suivant, l'historique du système a changé, donc la commande n'est pas le seul paramètre variant.

$$\frac{\partial y_s(k+1)}{\partial u(k)} \approx \frac{y_s(k+1) - y_s(k)}{u(k) - u(k-1)} \quad (4.27)$$

Une solution préférable est alors d'évaluer le jacobien en ligne avec un modèle non linéaire du pendule, identifié hors ligne, de façon suffisamment précise. Par exemple, l'utilisation d'un modèle neuronal direct conduit à la structure de régulation indirecte décrite dans le paragraphe (4.1.2). Cette architecture permet de reconstruire les jacobiens nécessaires, [JOR,89] [NAR,90].

Un modèle neuronal direct permet de faire l'évaluation du jacobien à chaque instant d'échantillonnage, d'une autre façon, en appliquant de petites variations de commande et mesurant les sorties résultantes du système, voir la figure (4.10). La figure (4.11) indique également des montages, avec un modèle du pendule et un modèle du régulateur, permettant d'évaluer les expressions des formules (4.23) et (4.24), utilisées dans le cas d'adaptation à deux pas du régulateur neuronal. Dans ce cas, le modèle du régulateur doit être réactualisé après chaque phase d'adaptation, par recopie des poids du régulateur. Ainsi le modèle du régulateur permet d'évaluer, à différents instants, les dérivées partielles de la commande par rapport aux poids du réseau ou à ces entrées. Le modèle du système peut fournir les dérivées des sorties du système par rapport aux signaux historiques ou par rapport à la commande courante (jacobien). Enfin, les deux modèles mis en série génèrent les sorties du pendule à différents instants, en fonction des variations des poids du modèle du régulateur (cas de la figure considérée).

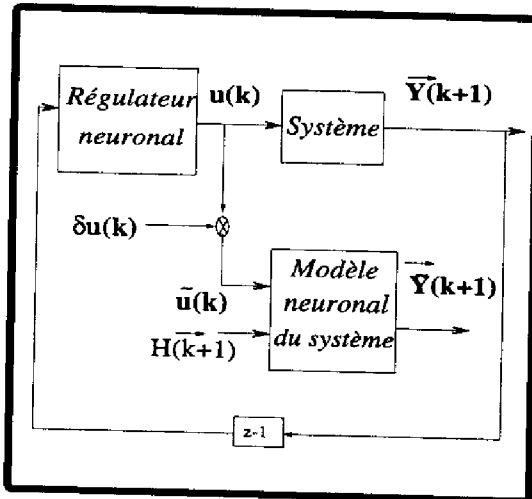


Figure 4.10 : Evaluation en ligne de J_s avec un modèle neuronal direct.

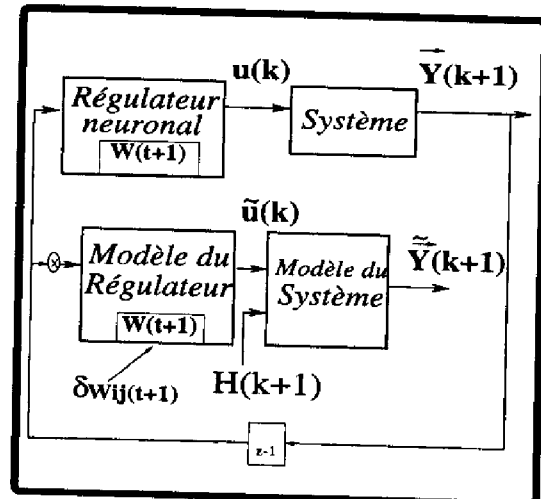


Figure 4.11 : Evaluation en ligne de diverses dérivées partielles.

Nous avons choisi, ici, la solution d'approximer le jacobien par les signes de ses composantes. Cette approche conduit à de bons résultats pour beaucoup de systèmes [SAE,91]. Elle ne conduit pas à une dégradation importante du calcul du gradient pour des systèmes mono-entrée/multi-sorties, tel que toutes les fonctions de sensibilité des variables de sortie aient le même signe.

En effet, les formules (4.28) mettent en évidence le fait que toutes les composantes du gradient sont proportionnelles au terme incluant les fonctions de sensibilité, et ceci quel que soit le critère utilisé. Les expressions des termes d'erreur $\xi_s^{[t_s+1]}(k+1)$, pour les critères proposé et classique, sont données respectivement en (4.29) et (4.30).

$$\left\{ \begin{array}{l} \vec{G}_r(k+1) = \{ \sum_{s=1}^4 \xi_s^{[s+1]}(k+1) \mathbf{J}_s(\mathbf{k}+1) \} \vec{R}(k+1) \\ \vec{R}(k+1) = f'[I_j^{[s]}(k)] \left(\begin{array}{c} u_i^{[s-1]}(k) \\ \vdots \\ \hline w_{ij}^{[s]}(k) f'[I_i^{[s-1]}(k)] u_h^{[s-2]}(k) \\ \vdots \end{array} \right) \end{array} \right. \quad (4.28)$$

$$\begin{aligned} \xi_s^{[s+1]}(k+1) &= \begin{cases} n_\theta \text{ erreur}_{\text{sortie}} & (s=1) \\ n_{\dot{\theta}} \text{ erreur}_{\text{sortie}} & (s=2) \\ n_y n_\theta \text{ erreur}_{\text{sortie}} & (s=3) \\ n_{\dot{y}} n_\theta \text{ erreur}_{\text{sortie}} & (s=4) \end{cases} \\ \text{erreur}_{\text{sortie}} &= n_\theta (\theta - \theta_{ref}) + n_{\dot{\theta}} \dot{\theta} \end{aligned} \quad (4.29)$$

$$\xi_s^{[s+1]}(k+1) = n_s^2 Y_s(k+1) \quad s \in [1, 4] \quad (4.30)$$

Dans le premier cas, à chaque itération, tous les termes d'erreurs $\xi_s^{[s+1]}(k+1)$ ont le même signe, ce qui n'est pas garanti dans le second cas.

Nous pouvons affirmer, dans le cas du critère proposé, que si toutes les composantes du jacobien ont le même signe pour toutes les sorties considérées, les signes de $\{ \sum_{s=1}^4 \xi_s^{[s+1]}(k+1) J_s(k+1) \}$ et $\{ \sum_{s=1}^4 \xi_s^{[s+1]}(k+1) \text{SIGNE}[J_s(k+1)] \}$ sont identiques. Dans ce cas, toutes les composantes du gradient sont modifiées dans les mêmes proportions, sans inversedement de signe. Ainsi, l'approximation effectuée sur le jacobien ne modifie pas la direction et le sens de convergence vers le minimum à chaque itération, mais seulement le pas de déplacement.

Par contre, dans le cas du critère des moindres carrés, avec des signes des composantes du jacobien identiques ou différents (ou dans le cas du critère proposé, avec des fonctions de sensibilité de signes différents pour les différentes sorties), la garantie de se déplacer à chaque itération dans le bon sens dans la direction du gradient n'est plus apportée.

Nous pouvons en déduire que l'approximation par le signe des composantes du jacobien ne pose pas de problème important avec notre critère, dans le cas de la régulation de l'angle (J_θ et $J_{\dot{\theta}}$ négatifs). Par contre, le cas de la régulation de l'angle et du chariot est plus délicat (J_θ et $J_{\dot{\theta}}$ négatifs, J_y et $J_{\dot{y}}$ positifs). L'approximation du jacobien par les signes de ses composantes peut conduire, dans ce dernier cas, à des problèmes d'apprentissage.

Dans le cas d'un problème multi-commandes, la direction de plus grande descente calculée à chaque instant d'échantillonnage est modifiée; l'approximation du jacobien est alors un problème beaucoup plus délicat.

4.2.4 Mise en oeuvre: choix des paramètres du réseau

Le problème essentiel posé par l'utilisation d'outils neuronaux est l'apprentissage effectif de la loi de régulation escomptée, lié à l'aspect convergence de l'apprentissage. En mode *apprentissage en ligne*, les actions de commande générées par le régulateur neuronal en fonction de la position dans le plan de phase, sont guidées par la procédure d'optimisation des poids du réseau. Localement, à chaque phase d'apprentissage, le réseau s'adapte de façon à ce que sa sortie se rapproche de la commande $u(k)$ permettant d'amener $\vec{Y}(k+1)$ à $\vec{0}$. Pour pouvoir retenir une loi d'asservissement, une commande adéquate doit être générée pour tous les vecteurs de sortie du système. Des preuves de stabilité au sens de *Lyapunov* existent (convergence des poids vers leur valeur optimale et de l'erreur vers 0) pour l'algorithme de *Jordan*, [SAE,93].

Beaucoup de réglages heuristiques caractérisent l'utilisation d'un réseau neuronal, même si certaines indications théoriques peuvent guider certains choix. Plusieurs facteurs déterminent la qualité de la convergence de l'algorithme d'apprentissage:

les paramètres physiques du réseau: ils sont essentiellement la structure du réseau et les caractéristiques des neurones. Au niveau des neurones, une fonction d'activation sigmoïdale conduira à une approximation globale de type généralisation, alors qu'une fonction à base radiale permet d'intégrer plus d'informations locales. Concernant la topologie des réseaux, le réseau neuronal de type Perceptron (*feedforward*), associé à la technique d'apprentissage par rétropropagation, est caractérisé par trois couches avec un nombre non pré-déterminé de neurones dans la couche cachée, voir le théorème de *Weierstrass* [NAR,90]. *Kolmogorov* a fourni des indices plus précis: $N(2N+1)$ neurones avec des fonctions d'activation non linéaires croissantes permettent d'approximer toute fonction continue de N variables, [LIP,87]. De plus, les études de [VIL,92] montrent que l'utilisation de plus de trois couches n'est pas nécessaire. La définition du réseau repose sur des choix basés sur l'expérience, mais des méthodes de construction automatique de la structure peuvent aussi être utilisées. Des procédures algorithmiques de diminution du nombre de connexions et de neurones, en fonction de leur efficacité, peuvent être mises en place. La plupart du temps, le réseau de neurones est utilisé en boîte noire et une structuration initiale spécifique n'est pas établie. Toutefois, dans certains cas, l'utilisation d'une connaissance initiale disponible permet de décomposer l'architecture globale en sous-systèmes, apportant alors plus d'efficacité.

la connaissance initiale intégrée dans les poids synaptiques: dans le cas général, elle est nulle (poids synaptiques initialisés à de petites valeurs aléatoires); mais elle peut être construite à partir de l'apprentissage d'une loi de régulation linéaire.

type d'algorithme d'apprentissage: la procédure classique de rétropropagation possède le désavantage d'être assez longue à converger, avec des risques d'arrêt sur des minima locaux. D'autres méthodes de type statistiques, telles le recuit

simulé sont encore plus lentes, mais plus sûres de converger vers le minimum global.

la qualité des approximations effectuées: les estimations des jacobiens et la fréquence de modification des poids peuvent entraîner des écarts dans le calcul de la direction et dans le pas du gradient.

les paramètres d'apprentissage: les taux d'apprentissage α et facteur de moment γ , choisis heuristiquement entre 0 et 1, influent sur l'importance des modifications à chaque étape.

la base d'exemples: elle doit être suffisamment complète par rapport au nombre de neurones dans la couche cachée pour que l'algorithme converge. Elle ne doit pas contenir des éléments trop corrélés entre eux, pour éviter de tendre vers des minima locaux.

le critère: il définit la forme de la courbe d'erreur, donc l'existence éventuelle de minima locaux.

4.3 Application au pendule inversé

Des expériences en simulation et sur site réel ont été réalisées, [FUT,93] [DEG,93] [DEG,94a] [DEG,95a].

4.3.1 Simulations

A- Contexte des simulations

Des régulateurs neuronaux ont déjà été utilisés pour stabiliser le pendule inversé, voir [SAE,91]. Une première évaluation expérimentale de la structure de régulation neuronale est menée ici sur les modèles sans frottements, décrits dans le paragraphe (1.4.1) du chapitre (1). Les modèles utilisés, à 2 puis 4 sorties, ont pour paramètres physiques ($l = 52cm$, $M = 1.4Kg$, $m = 0.37Kg$).

Les expériences menées ont pour but de vérifier que la technique d'adaptation des poids du réseau de neurones permet d'atteindre la stratégie de régulation définie par le critère d'apprentissage. Pour cela, sont observées les trajectoires dans le plan de phase pour certains facteurs de pondération du critère (permettant de fixer les performances dynamiques). Le problème est, ensuite, de déterminer si une loi de commande est effectivement intégrée dans le régulateur neuronal. Pour cela, après la phase d'apprentissage, des tests de régulation sont menés sans adaptation.

Tout d'abord, le schéma du régulateur neuronal mis en oeuvre est celui de la figure (4.4) du paragraphe (4.2.1). Dans le deuxième cas, le schéma du régulateur neuronal mis en oeuvre est proposé sur la figure (4.5) du paragraphe (4.2.1). Dans les deux cas, la topologie du réseau utilisé est composée de trois couches avec huit neurones dans la couche cachée. Nous avons choisi des fonctions d'activation sigmoïdes.

En ce qui concerne la mise en oeuvre de l'apprentissage, les facteurs d'apprentissage sont d'abord choisis:

régulation de θ : $\alpha = 0.1$, $\gamma = 0.05$;

régulation de θ et y : $\alpha = 0.2$, $\gamma = 0.1$.

Ensuite, les facteurs du critère sont fixés:

cas 1: $n_\theta = 1$, $n_{\dot{\theta}} = 2$;

cas 2: $n_\theta = 1$, $n_{\dot{\theta}} = 2$, $n_y = 1$, $n_{\dot{y}} = 3$.

Après avoir choisi les spécifications en boucle fermée, des CI pour l'angle sont choisies aléatoirement dans le domaine $[-\theta_l, \theta_l]$ ($\theta_l = 0.4\text{rad}$ dans le premier cas et 0.5rad dans le second cas). Pour chaque CI angulaire, les autres composantes du vecteur de sortie sont choisies nulles. m simulations de régulation avec adaptation des poids du réseau sont alors lancées jusqu'à la condition de fin (horizon de simulation écoulé -respectivement 5s et 16s- ou échec -tige ou chariot ayant dépassé une position limite-). Au bout de m simulations, (m vaut 100 dans le premier cas et 840 dans le second cas), une nouvelle CI angulaire est choisie.

L'apprentissage est, par la suite, évalué sur des données particulières (pour une CI particulière dans le domaine de fonctionnement angulaire), constituant une base de validation. Notre critère de performance est la convergence vers zéro de l'erreur de sortie du pendule, au cours du test de régulation, et le suivi de la droite de glissement pré-fixée.

B- Résultats avec le modèle à deux sorties

Les résultats de la phase de test sans apprentissage sont présentés sur les figures (4.12) à (4.14), représentant respectivement la réponse angulaire, les commandes utilisées et la trajectoire dans le plan de phase $\dot{\theta}(\theta)$. Le temps de réponse est 2.5s et la pente de la droite de glissement est pratiquement égale à -2 , comme attendu. Ces courbes montrent la possibilité d'apprendre une stratégie de régulation fixée par le critère d'apprentissage. La courbe (4.15) présente le parcours du plan de phase lorsqu'une simulation est ensuite effectuée avec apprentissage. Nous pouvons constater une adéquation totale avec la trajectoire escomptée, attestant d'un apprentissage non parfait dans le cas précédent.

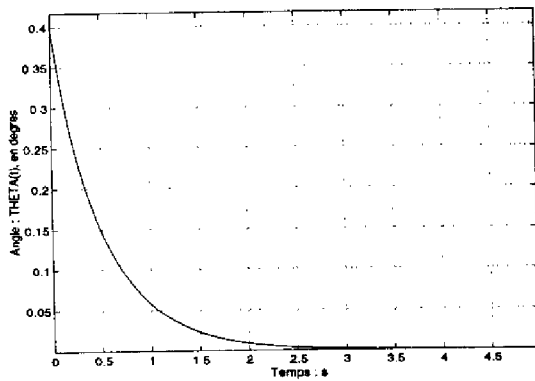


Figure 4.12 : Réponse angulaire.

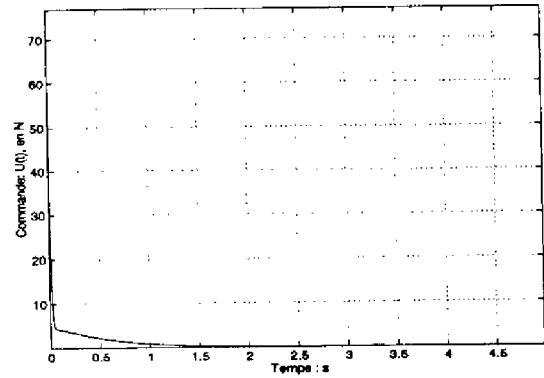
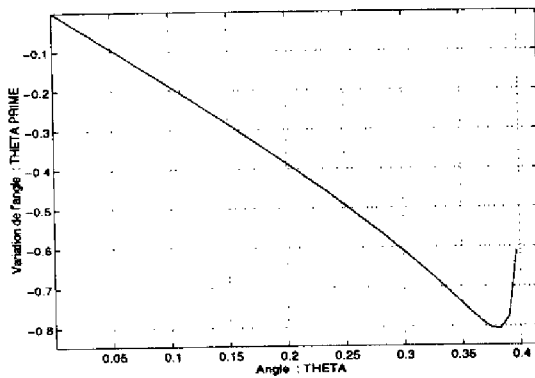
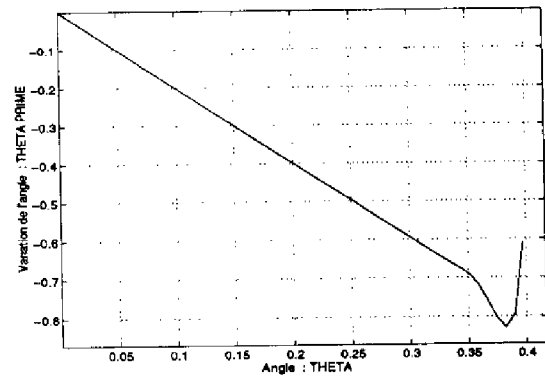


Figure 4.13 : Commande.

Figure 4.14 : $\dot{\theta}(\theta)$ sans adaptation.Figure 4.15 : $\dot{\theta}(\theta)$ avec adaptation.

C- Résultats avec le modèle à quatre sorties

Les résultats de la phase de test, après l'apprentissage, sont fournis sur les figures (4.16) à (4.19), qui représentent respectivement la réponse angulaire avec le signal de référence angulaire, le plan de phase $\dot{\theta}(\theta - \theta_{ref})$, la position horizontale du chariot et la vitesse de déplacement horizontal. Le temps de réponse à 0% de l'angle est d'environ 9s, celui de la position du chariot est de 13s. La première figure permet de souligner que la référence angulaire est suivie correctement durant la phase de régulation et permet d'expliquer les oscillations de l'angle. La trajectoire suivie dans le plan de phase pour atteindre la position-but montre que l'état du système peut rejoindre la droite de glissement de pente (-2), voir la figure (4.17). La trajectoire hésitante, au début dans le plan de phase, peut être liée à la qualité de l'apprentissage (durée, exemples utilisés, approximations faites..).

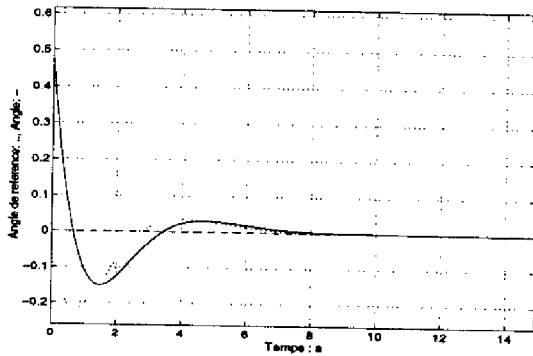
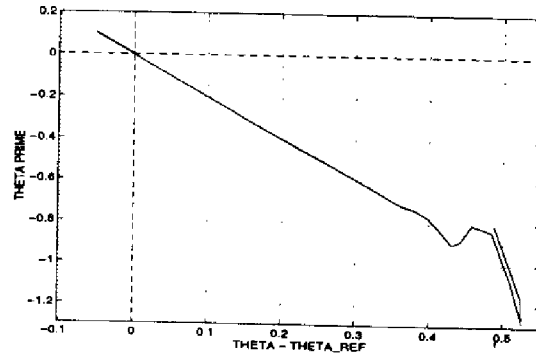
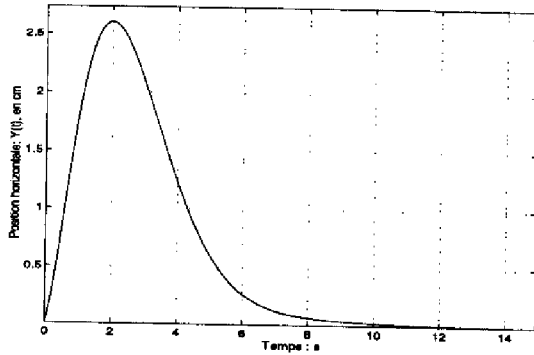
Figure 4.16 : $\theta(k)$ (-) et $\theta_{ref}(k)$ (..).Figure 4.17 : Plan de phase: $\dot{\theta}(\theta - \theta_{ref})$.

Figure 4.18 : Position horizontale.

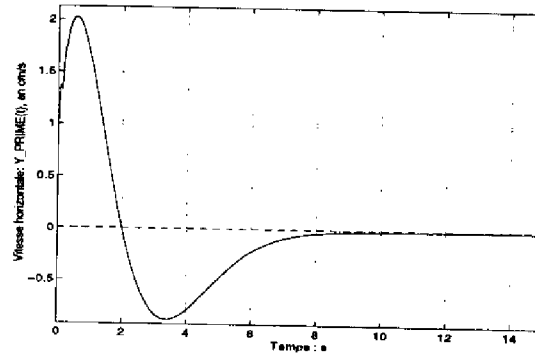


Figure 4.19 : Vitesse horizontale.

4.3.2 Système réel

A- Contexte des expérimentations

Le pendule inversé est décrit dans le paragraphe (1.4.1) du chapitre (1). Les valeurs des paramètres physiques considérés sont ($l = 52\text{cm}$, $M = 1.4\text{Kg}$, $m = 0.37\text{Kg}$). Le schéma du régulateur neuronal mis en oeuvre est proche de celui proposé sur la figure (4.5). La différence est une entrée supplémentaire ajoutée pour introduire directement la possibilité de faire de la compensation de frottements.

En ce qui concerne l'apprentissage sur site réel, le système considéré ayant une dynamique assez rapide, des problèmes de durée d'apprentissage peuvent se poser. En effet, à l'intérieur d'une période d'échantillonnage, doivent se dérouler à partir de l'instant $k(T_e)$: la restitution d'une commande par le réseau, l'application de $u(k)$ au système, l'acquisition des signaux de mesures $\vec{Y}(k+1)$ et l'adaptation des poids du réseau. Disposant d'un PC rapide (486DX2, 66Mhz), nous avons pu utiliser douze neurones dans la couche cachée.

Les régulations temps réel de θ et de y , puis de θ seulement, sont réalisées avec des données venant du système réel. Un filtre du premier ordre est appliqué sur les signaux θ et y , les vitesses $\dot{\theta}$ et \dot{y} étant déterminée numériquement.

Dans le premier cas, trente essais avec apprentissage sont menés avant d'obtenir les résultats présentés ci-dessous (avec pour conditions d'échec $\theta = 0.5\text{rad}$ ou $y = 50\text{cm}$). Les paramètres d'apprentissage sont $\alpha = 0.02$ et $\gamma = 0.01$. L'objectif des expériences est de stabiliser la tige du pendule. En effet, reproduire le résultat obtenu en simulations est quasiment impossible, puisqu'il faudrait remettre la tige en position haute manuellement, environ 1000 fois.

Dans le deuxième cas, seulement cinq itérations d'apprentissage sont réalisées.

B- Résultats des régulations de θ et y

Les courbes (4.20) à (4.23) présentent successivement, pour une régulation avec adaptation, la réponse angulaire avec la référence angulaire, le signal de commande, la position et la vitesse du chariot. Nous pouvons noter une capacité du régulateur neuronal à faire converger la tige et le chariot vers leur position-cible, malgré une non initialisation préalable du contenu du réseau et une estimation grossière du gradient.

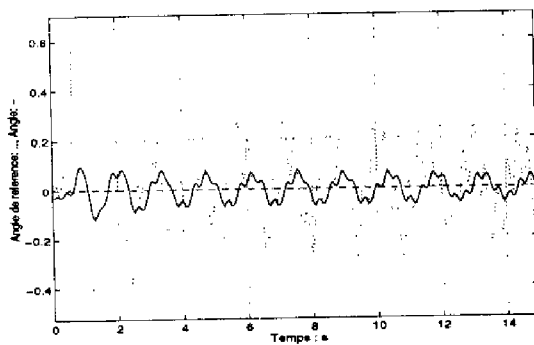


Figure 4.20 : $\theta(k)$ - et $\theta_{ref}(k)$.. rad.

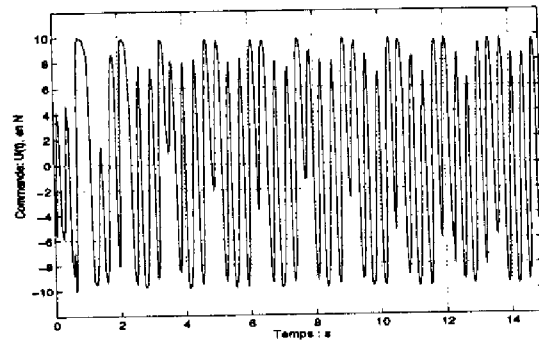


Figure 4.21 : Commandes.

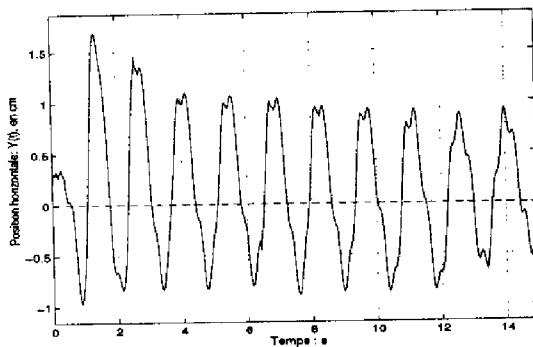


Figure 4.22 : Position horizontale.

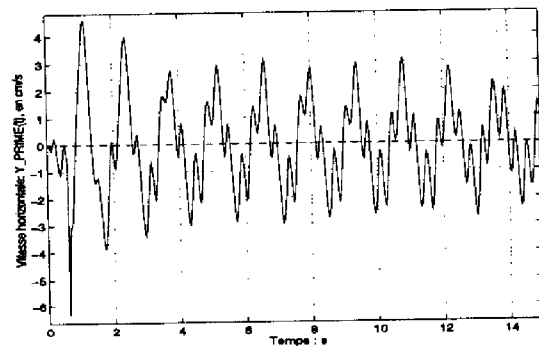


Figure 4.23 : Vitesse horizontale.

Nous devons toutefois noter les difficultés de réglage des nombreux paramètres du régulateur neuronal et de validation de la stratégie de régulation à régime glissant sur le système réel. Outre le choix des facteurs d'apprentissage, qui doivent être suffisamment faibles pour ne pas empêcher la convergence vers la droite de glissement, il est nécessaire de choisir une dynamique atteignable par le système instable (facteurs n_θ , $n_{\dot{\theta}}$, n_y , $n_{\dot{y}}$ et les saturations sur θ_{ref}). Le problème suivant est d'obtenir une bonne convergence de l'algorithme d'apprentissage vers les performances dynamiques choisies. Cet aspect est délicat (à cause des effets de l'approximation par les signes du Jacobien et de la démarche manuelle de réinitialisation de la position de la tige). C'est pourquoi, il est préconisé d'initialiser la connaissance interne du réseau avec les résultats de l'apprentissage en simulation, même si le modèle est différent, ou avec un algorithme de régulation linéaire.

C- Résultats des régulations de θ

La courbe (4.24) montre la stabilisation quasi-immédiate de l'angle, après cinq départs de la position haute seulement. Dans ce cas, l'adaptation est très efficace et la procédure manuelle d'apprentissage aisée à réaliser.

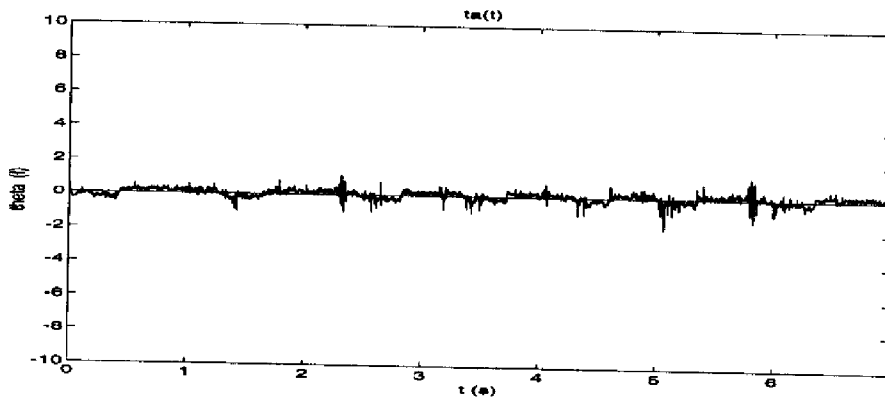


Figure 4.24 : Réponse angulaire en degrés.

4.4 Conclusions et extensions

A- Conclusions

L'apprentissage implicite d'une loi d'asservissement de type PD est possible, à partir de données expérimentales et d'une connaissance a priori quasi-nulle sur le système

et sur la structure du régulateur. La méthodologie de conception proposée permet de fixer, par le critère d'apprentissage, la stratégie d'évolution dans le plan de phase, ou, en d'autres termes, la vitesse de convergence vers la consigne.

Les résultats obtenus, durant les expériences avec des modèles simulés du pendule, soulignent la capacité du régulateur neuronal à apprendre la stratégie de régulation choisie par l'utilisateur. Toutefois la précision de la stratégie de régulation reproduite dépend de la qualité de l'apprentissage.

Les expériences menées sur le système réel, dans le cas des régulations de θ et y , montrent une capacité du régulateur neuronal à stabiliser le pendule inversé. Toutefois des difficultés à mettre en oeuvre l'apprentissage sont apparues. Dans le cas de la régulation de la tige seulement, la stabilisation est immédiate.

B- Extensions

Il est, tout d'abord, nécessaire de finir les évaluations sur le site réel afin d'observer si, dans ce cas, une loi d'apprentissage peut être apprise (avec pré-initialisation de la connaissance du réseau à partir d'un régulateur linéaire), et ensuite de mesurer expérimentalement la robustesse en stabilité. Il serait également intéressant de réaliser des tests en simulation avec d'autres paramètres du régulateur, permettant d'optimiser les performances dynamiques en boucle fermée, afin de pouvoir comparer les résultats obtenus avec ceux du régulateur flou du paragraphe (3.1.4) du chapitre (3).

L'évaluation du RN étant achevée, des modifications de la démarche de conception du régulateur pourraient être envisagées. La procédure d'apprentissage pourrait être améliorée en utilisant des expressions du gradient plus précises, dans le but d'avoir une réponse dans l'espace d'état plus proche de ce qui est exprimé par le critère d'optimisation. Deux voies sont exploitables: adapter les poids du réseau tous les trois instants d'échantillonnage pour calculer une meilleure direction de descente globale, ou utiliser d'autres façons d'approximer le jacobien, voir le paragraphe (4.2.3). Ce dernier aspect est particulièrement essentiel dans le cas de la régulation de θ et y , pour lesquels l'approximation par les signes peut entraîner des erreurs locales d'apprentissage.

D'autres spécifications de performances dynamiques pourraient également être définies via le choix d'autre trajectoires de convergence dans le plan de phase, en définissant d'autre fonctions d'erreur pour la procédure de rétropropagation. Ces modifications pourraient conduire à générer une loi de régulation plus fortement non linéaire.

Concernant les temps de calculs nécessaires au RN pour l'adaptation des poids, des mises en oeuvre parallèles seraient envisageables.

Nous abordons maintenant des structures de régulation plus complexes, incluant des modules flous et des approches conventionnelles....

Chapitre 5

Réglage d'un PID par logique floue

Afin de pouvoir bénéficier des avantages fournis par les régulateurs algorithmiques et par les techniques considérées dans ce travail, il est intéressant d'évaluer des combinaisons des différentes approches. Aussi, la dernière étape de notre travail consiste à régler optimalement les paramètres de régulateurs, afin d'améliorer les performances des systèmes en boucle fermée.

L'approche considérée ici est la *supervision floue* de régulateurs PID. La loi de régulation initiale, de type PI analytique, est améliorée avec des techniques floues. Ainsi, le point de départ du réglage est une base connue, largement utilisée dans l'industrie. Mais bien que le PID soit robuste dans des conditions de fonctionnement nominales, il nécessite des ajustements en raison de performances parfois inacceptables, voir le paragraphe (1.1.2) chapitre (1).

Nous avons développé deux superviseurs flous de même structure initiale, mais avec des réglages différents, dont certains apportent des garanties de stabilité. Des comparaisons sont menées entre les régulateurs PID supervisé, PID non supervisé et PID industriel sur les fours.

Dans ce chapitre sont présentés: un état de l'art des superviseurs flous (5.1), les modules adaptatifs développés (5.2), les résultats d'expérimentations (5.3) et, enfin, les conclusions sur ces structures de commande (5.4).

5.1 Définition des superviseurs flous

Malgré la dénomination *supervision floue* souvent utilisée dans la littérature, nous considérons, en fait, un adaptateur expert d'un régulateur PID.

5.1.1 Principe général

Les premières utilisations de systèmes experts, pour superviser des régulateurs PID, remontent au début des années 1980. Une liste succincte de superviseurs experts de PID est proposée ci-après:

- 1984 - *Kraus & Myron*: les paramètres du PID sont ajustés de façon à obtenir une réponse plus rapide, tant que le dépassement et le facteur d'amortissement limites ne sont pas dépassés;
- 1985 - *Astrom*: le superviseur est utilisé pour choisir le meilleur algorithme numérique de régulation en fonction des spécifications de l'utilisateur;
- 1988 - *Devanathan & al*: le superviseur calcule, durant chaque régime transitoire, les paramètres du système et l'erreur sur les performances atteintes, afin de pouvoir régler optimalement les paramètres du PID;
- 1990 - *Tzafestas & Papanikolopoulos*: le superviseur utilisé est un système expert flou.

Les raisons principales pour utiliser un PID dans une boucle de régulation sont une maîtrise industrielle de ce type de régulateur et des garanties possibles de stabilité. D'autre part, les superviseurs de PID sont facilement mis en oeuvre. En effet, peu de connaissance a priori sur le système à réguler est nécessaire; un modèle mathématique n'est, par exemple, pas requis. A l'opposé, les régulateurs adaptatifs conventionnels nécessitent souvent la connaissance de la structure du modèle du processus régulé, [OLI,91]. Un autre avantage des superviseurs est l'interface offerte à l'utilisateur pour exprimer ses souhaits, quant aux performances en boucle fermée, [NAU,90] [TZA,90]. Le choix d'un type de superviseur à base de règles, et plus précisément avec un formalisme flou, amène de multiples avantages (simplicité, souplesse, adaptabilité....), décrits dans le paragraphe (1.2.2) du chapitre (1).

La structure interne du superviseur est identique à celle d'un régulateur flou, voir le paragraphe (2.1.2) du chapitre (2). Les paramètres principaux du superviseur flou sont les facteurs d'échelle (liés aux plages de variation réelle des données en entrée du superviseur et des trois paramètres du régulateur PID en sortie), les ensembles flous sur les différents univers de discours normalisés et le contenu des règles d'adaptation des paramètres du PID.

5.1.2 Les deux approches principales

Dans la littérature, il est possible de dégager deux démarches concernant la supervision floue d'un régulateur PID. Les sorties du superviseur flou sont les variations à appliquer aux trois gains du PID. Les entrées peuvent être de deux types:

- Dans le premier cas, les entrées du superviseur sont l'erreur de sortie du système et sa dérivée. Les adaptations menées sur les gains du PID visent à corriger la loi de régulation au fur et à mesure de l'évolution du système, à chaque instant d'échantillonnage.
- Dans le deuxième cas, les entrées du superviseur sont les performances atteintes par le système en boucle fermée, au cours du régime transitoire précédent. La mesure de la qualité de la régulation, au cours d'un certain intervalle temporel, permet de décider d'un changement sur la loi de commande, de façon à obtenir de meilleures performances dans le futur.

A- Supervision d'un PID à partir de l'erreur de sortie et de sa dérivée

Il est possible de distinguer les méthodes qui consistent en des adaptations complètement indépendantes des trois gains du PID et celle qui proposent des adaptations liées.

Dans le premier cas, trois bases de règles constituent le superviseur flou; elles définissent les degrés de variation à appliquer aux trois gains K_p , K_i et K_d , en fonction des valeurs de e et \dot{e} . Les actions sur les gains sont complètement décorrélées et des nuances complexes peuvent être introduites. En contrepartie, la phase de réglage des paramètres peut s'avérer fastidieuse.

Il est possible de citer comme application de cette approche [BAB,92] et [NAU,90]; dans ce dernier cas, une base de connaissance permet un réglage automatique du superviseur.

Dans le deuxième cas, une unique base de règles sert à établir la stratégie d'évolution des trois gains du PID. Les changements sur chacun des paramètres sont calculés en multipliant le résultat de l'inférence par un des trois coefficients k_1 , k_2 ou k_3 . Ces trois facteurs sont liés aux domaines de variation des gains du PID; ils sont déterminés par des analyses de stabilité et par des choix de caractéristiques en boucle fermée (par exemple: préférence donnée à la rapidité au détriment des oscillations...), [TZA,90]. Des applications de cette méthode sont proposées dans [TZA,90], où le superviseur utilise la matrice floue de *MacVicar-Whelan*.

Dans les deux cas, la logique globale d'adaptation du PID repose sur les liens entre l'évolution des paramètres du régulateur et les performances en boucle fermée. *Loin de la référence*, il faut augmenter K_p et K_i ($\frac{1}{T_i}$) pour réduire le temps de montée; *près de la référence*, il faut augmenter K_d (T_d) pour diminuer les oscillations. Les variations des gains proportionnel et intégral ont des influences sur la dynamique du système qui varient dans le même sens. En fonction de la position dans le plan de phase, il s'agit de mener des actions, qui se basent sur les tendances citées précédemment et sur les souhaits de l'utilisateur en termes de performances.

B- Supervision de PID à partir des performances dynamiques atteintes

La qualité des performances atteintes peut-être mesurée par le premier dépassement, la période des oscillations, le temps de montée.... De nouveau, deux types de mise en oeuvre sont réalisables.

La première mise en oeuvre consiste à tenir compte de toutes les influences entre les trois gains du PID et les performances dynamiques mesurées. [LIT,91] et [KET,92] ont proposé des exemples de mise en oeuvre. Dans le premier cas, l'utilisateur exprime ses spécifications en attribuant des poids à chaque performance désirée. Le problème est de gérer les influences parfois contradictoires entre les variations des différents gains du PID et les évolutions de la réponse temporelle. Les définitions de valeurs limites non acceptables, pour les diverses caractéristiques dynamiques, et de règles spéciales de gestion des priorités de réglage apportent une solution. Le paramètre du PID qui est réglé en premier est celui qui entraîne une réduction du dépassement, puis celui qui influe sur le moins de performances dynamiques en même temps et, enfin, celui qui permet d'améliorer la stabilité. Dans [KET,92], les modifications de chaque gain du PID sont une combinaison linéaire des variations de ces gains, déduites de chaque performance dynamique mesurée, voir aussi [KET,95].

La deuxième mise en oeuvre est une simplification, consistant à ne considérer que les relations dominantes. [OLI,91] propose une telle approche. Les liens prédominants entre K_p et t_m , d'une part, et entre K_i et D , d'autre part, sont considérés pour adapter un régulateur PI. Les deux bases de règles utilisées tendent à faire converger les caractéristiques dynamiques vers leurs valeurs floues *Moyennes*. Les définitions de ces ensembles flous permettent donc de définir un modèle de référence.

Dans les deux cas, les règles d'adaptation peuvent être déduites d'un ensemble de réponses temporelles obtenues en simulation pour différents couples (K_d, K_i) pour chaque valeur de K_p , [LIT,91]. Les courbes permettent de mettre en évidence des relations qualitatives, telles celles détaillées ci-après. *Si K_p ou K_i augmente*, alors t_m (temps de montée) diminue, D (premier dépassement) et Osc (amplitudes des oscillations) augmentent; *si K_d augmente*, alors D et Osc diminuent.

5.2 Mises en oeuvre des superviseurs flous

5.2.1 Régulateurs PID et superviseur considérés

A- Equations des régulateurs PID considérés pour ce travail

Elles sont décrites en (5.1) pour les simulations et (5.2) pour les expériences temps réel avec le régulateur industriel; $C(k)$ est la valeur courante du signal de référence et T_e est la période d'échantillonnage.

Le régulateur industriel possède un certain nombre de particularités par rapport au régulateur PID académique. Des actions de saturations et désaturation du gain

intégral sont utilisées: l'action intégrale est activée lorsque la réponse entre dans la bande proportionnelle (la bande proportionnelle $-B_p-$ est définie à partir du gain proportionnel, voir l'équation (5.3) $-E$ est l'échelle de sortie du régulateur-). Elle est maximale au-dessus de cette zone et est nulle en dessous. Ceci permet d'éviter des valeurs de commande trop élevées pour les erreurs importantes. Une autre fonction spécifique est la fonction *CAS*, introduite par la société *CORECI*. Un terme de commande directe est ajouté au signal généré par le régulateur, voir les équations (5.2) et (5.4) (C_p est le niveau de consigne en régime permanent, ϵ est l'erreur en régime permanent obtenue avec un régulateur proportionnel). La correction *CAS* est un dispositif breveté, utilisé avec un régulateur proportionnel dans le cas d'un système linéaire. Il permet d'annuler exactement l'erreur statique pour une consigne donnée. Dans le cas de la régulation d'un système non linéaire ou avec perturbations externes, on ajoute une action intégrale pour compenser les faibles écarts en sortie du processus et moduler plus finement la puissance fournie.

$$U_{PID}(k) = U(k-1) + K_p \left\{ e(k) \left[1 + \frac{T_e}{T_i} + \frac{T_d}{T_e} \right] - e(k-1) \left[1 + 2\frac{T_d}{T_e} \right] + e(k-2) \left[\frac{T_d}{T_e} \right] \right\} \quad (5.1)$$

$$U(k) = U(k-1) + U_{PID}(k) + U_{CAS}(k) \quad U_{CAS}(k) = CAS \times C(k) \quad (5.2)$$

$$B_p = \frac{100}{K_p} \quad \text{et} \quad X_p = \frac{B_p \times 100}{E} \quad (5.3) \quad CAS = \frac{10^4 \times \epsilon}{C_p \times X_p} \quad (5.4)$$

Les gains initiaux du PID sont déduits de la méthode du relais, sur un niveau de consigne intermédiaire dans la plage des consignes considérée, et des formules de *Ziegler* et *Nichols* pour les échelons et *Ziegler* et *Nichols* adaptées pour les rampes, voir le paragraphe (3.2.1) du chapitre (3).

Pour un PID sans fonction *CAS*, un signal de référence modifié permet d'obtenir un meilleur suivi de consigne en rampe. Le but est d'atténuer le dépassement qui apparaît après la transition entre le rampe et le plateau constant. La forme exacte du signal de référence étant connue à l'avance (pente et niveau de consigne pré-fixés), une sous-référence est définie, voir la figure (5.1).

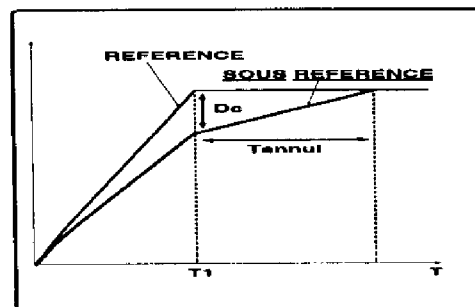


Figure 5.1 : Définition d'une sous-référence.

La distance D_c entre le point de cassure de la référence et la position souhaitée pour la réponse temporelle étant choisie, il reste à proposer un temps d'annulation

de l'erreur (T_{annul}). Le mécanisme mis en place revient à effectuer une correction progressive en nuancant la réponse grâce à une sous-consigne. Lorsque la fonction CAS est utilisée, ce mécanisme supplémentaire devient inutile, car le CAS tient directement compte de la référence.

B- Principe de fonctionnement du superviseur

Le système expert flou utilisé possède des caractéristiques *standards*. Les opérateurs mathématiques utilisés par le moteur d'inférence sont respectivement: la conjonction de *Mamdani* (*minimum*), la règle compositionnelle d'inférence de *Zadeh* (*Sup_min*), le *minimum* pour le *et* et le *maximum* pour l'*agrégation* des règles. La méthode de défuzzification par moyenne pondérée est utilisée. L'inférence est réalisée en ligne compte tenu des constantes de temps des fours considérés (la plus courte est de 100 secondes).

La méthode de supervision floue développée a été inspirée de la méthode de *Tzafestas*, voir le paragraphe (5.1.2). C'est une stratégie du premier type, pour laquelle une seule base de règles, de variable de sortie α , assure la gestion des lieux du plan de phase où des actions d'adaptations doivent être menées. Des transformations d'échelle pour les trois gains du PID, dans leur univers de discours réel respectif, permettent de gérer l'importance relative donnée au réglage de chaque gain du régulateur. La figure (5.2) illustre le fonctionnement du superviseur flou considéré.

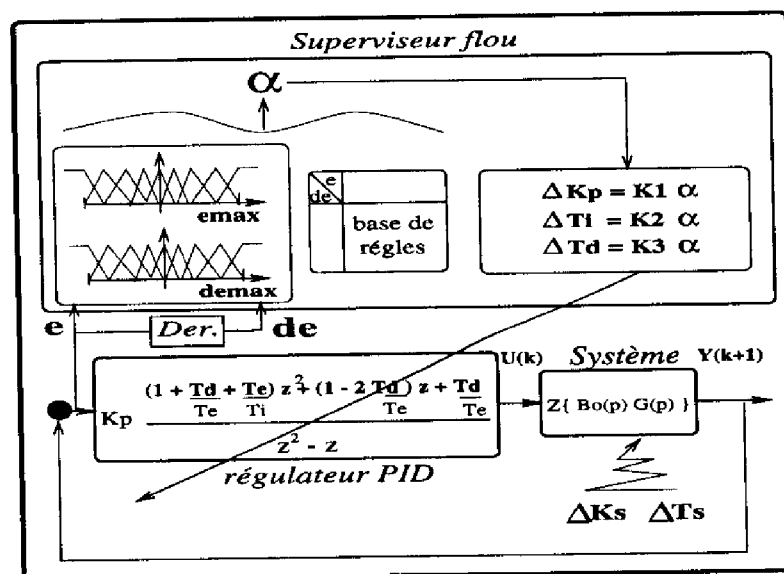


Figure 5.2 : Mise en oeuvre du superviseur flou.

Pourquoi avoir choisi un tel type de superviseur ? Tout d'abord, une contrainte industrielle était de faire une adaptation des paramètres du PID à chaque instant

d'échantillonnage, en fonction de l'erreur de sortie du système et de sa variation. Cette approche permet d'améliorer la réponse temporelle durant tout le régime transitoire. De plus, la méthode de Tzafestas n'utilise qu'une seule table de règles, ce qui peut être intéressant pour minimiser le temps de calcul (lors de l'inférence en ligne) et la complexité de l'étape de formalisation de la connaissance nécessaire au raisonnement. En outre, la logique de supervision étant exprimée par les règles et par les formules d'adaptation des paramètres du PID, le choix de ces formules permet d'avoir trois actions pondérées différentes sur les gains du PID et de prendre tout de même en considération les spécifications de l'utilisateur en terme de dynamique en boucle fermée.

Les paragraphes suivants décrivent deux stratégies de supervision proposées et présentent les choix effectués pour les paramètres du SF. Les systèmes à réguler sont le modèle du second ordre des fours avec gain variable et le simulateur non linéaire, voir le paragraphe (1.4.2) du chapitre (1). Dans un premier temps, les non linéarités dues au gain variable sont compensées (à partir d'analyses de mesures), ce qui conduit à un système linéaire. Ensuite, la régulation non linéaire, par adaptation floue, permet de faire de l'optimisation locale des transitoires.

5.2.2 Stratégies d'adaptation utilisées pour les gains du PID

A- Première approche

La table de règles de *MacVicar-Whelan*, reproduite dans le tableau (5.1), synthétise les actions de supervision.

<i>e de</i>	NG	NM	NP	NTP	ZE	PTP	PP	PM	PG
NG	NG	NG	NG	NG	NG	NM	NP	NTP	ZE
NM	NG	NG	NG	NG	NM	NP	NTP	ZE	PTP
NP	NG	NG	NG	NM	NP	NTP	ZE	PTP	PP
NTP	NG	NG	NM	NP	NTP	ZE	PTP	PP	PM
ZE	NG	NM	NP	NTP	ZE	PTP	PP	PM	PG
PTP	NM	NP	NTP	ZE	PTP	PP	PM	PG	PG
PP	NP	NTP	ZE	PTP	PP	PM	PG	PG	PG
PM	NTP	ZE	PTP	PP	PM	PG	PG	PG	PG
PG	ZE	PTP	PP	PM	PG	PG	PG	PG	PG

Tableau 5.1 : Table de règles de supervision donnant α .

Des actions de supervision non nulles sont menées lorsque la sortie du système n'est pas confondue avec la référence ou lorsque la vitesse de convergence n'est pas celle attendue. Plus précisément, dans des situations de correction importante (ex.: loin de la référence en début de régime transitoire -e est PG- avec une vitesse de convergence

insuffisante -de est NP-), la puissance est amplifiée par des augmentations de K_p et K_i . Au contraire, dans des situations de convergence trop rapide, ou au-delà de la référence, après la première intersection avec la consigne, K_p et K_i sont diminués et K_d est augmenté. Les formules (5.5) traduisent ces évolutions.

$$\begin{aligned} K_p(k+1) &= K_p(k) + \alpha \times K_1 & T_d(k+1) &= T_{d_{initial}} - \alpha \times K_3 \\ T_i(k+1) &= T_{i_{initial}} - \alpha \times K_2 & CAS(k+1) &= K_4 \times \frac{e}{C} \times K_p \end{aligned} \quad (5.5)$$

L'adaptation du paramètre CAS a lieu chaque fois que le gain proportionnel est adapté. Les adaptations sont, ici, de type incrémental pour le gain proportionnel, contrairement aux deux autres gains qui varient autour de leur valeur initiale. En effet, l'adaptation de la valeur de K_p permet de compenser la perturbation principale imposée par le cahier des charges, qui est l'évolution du gain du four en fonction de la référence. Cette adaptation permet effectivement de faire face à la perturbation interne sur le gain du four car, par exemple, si le gain du système K_s diminue par rapport à sa valeur nominale, le processus sera plus lent. Alors le nombre d'actions positives sur K_p seront nombreuses afin que la sortie du four rejoigne plus vite la consigne. Ainsi, l'évolution de la dynamique du régulateur permet de s'adapter à celle du four et de pouvoir converger vers les réglages optimaux pour chaque niveau de consigne. Les adaptations des valeurs de T_i et de T_d permettent d'optimiser localement les régimes transitoires, ces deux paramètres variant autour de leurs valeurs initiales.

Les paramètres K_1 , K_2 , K_3 et K_4 sont déterminés, ici, de façon à exprimer au mieux les spécifications de l'utilisateur concernant la dynamique du four régulé. Par exemple, si le comportement en boucle fermée désiré est une réponse rapide, sans condition très restrictive sur les dépassements, les paramètres K_2 et K_1 sont choisis grand devant K_3 .

Lorsque la sortie du four aborde une rampe descendante après un plateau, la réponse évolue au-dessus de la rampe et peut diverger de façon importante. Des actions spécifiques sont alors menées. En phase de divergence ($e < 0$, $de < 0$), des actions *Négatives* sont préconisées en sortie du superviseur -cases en gras de la table de règles (5.1)-. Dans le cas d'une rampe descendante, les signes des valeurs de α sont inversés en sortie du moteur d'inférence. Cette adaptation spécifique, qui permet d'améliorer le suivi de rampe descendante, met en lumière que le fait d'augmenter la dynamique du régulateur lors de la phase de divergence permet de moduler la commande plus rapidement, donc de façon plus adéquate. Ceci nous conduit à proposer la deuxième stratégie de supervision floue...

B- Deuxième approche incluant des garanties de stabilité

La dynamique du régulateur est maintenant systématiquement augmentée lorsque la sortie du système diverge ou ne converge pas assez vite. Au contraire, des actions de supervision négatives sont appliquées lorsque la réponse converge trop vite. La table

de règles (5.2) et les formules (5.6) traduisent globalement cette stratégie -les labels linguistiques soulignés ont la signification suivante: PG= Positif très Grand-. Par exemple, pour e : PM et de : PG (divergence) ou e PM et de NP (convergence trop lente), les actions de supervision positives générées se traduisent par des augmentations de K_p , K_i et par une diminution de K_d .

$$\begin{aligned} K_p(k+1) &= K_{p_{initial}} + \alpha \times K_1 \\ T_i(k+1) &= T_{i_{initial}} - \alpha \times K_2 \\ T_d(k+1) &= T_{d_{initial}} - \alpha \times K_3 \end{aligned} \quad (5.6)$$

$e \ de$	NG	NM	<u>NP</u>	<u>NP</u>	ZE	<u>PP</u>	PP	PM	PG
<u>NG</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>	PG	PM	NP	<u>NG</u>
<u>NM</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>	PG	PG	PM	NP	<u>NG</u>	<u>NG</u>
<u>NP</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>	PG	PM	NP	<u>NG</u>	<u>NG</u>	<u>NG</u>
<u>NP</u>	<u>PG</u>	<u>PG</u>	PG	PM	PP	NP	NM	NG	<u>NG</u>
<u>ZE</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>	PP	ZE	PP	<u>PG</u>	<u>PG</u>	<u>PG</u>
<u>PP</u>	<u>NG</u>	NG	NM	NP	PP	PM	<u>PG</u>	<u>PG</u>	<u>PG</u>
<u>PP</u>	<u>NG</u>	NG	<u>NG</u>	NP	PM	PG	<u>PG</u>	<u>PG</u>	<u>PG</u>
<u>PM</u>	<u>NG</u>	<u>NG</u>	NP	PM	PG	<u>PG</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>
<u>PG</u>	<u>NG</u>	NP	PM	PG	<u>PG</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>	<u>PG</u>

Tableau 5.2 : Deuxième stratégie de supervision donnant α .

Les adaptations des trois gains sont ici faites autour des valeurs initiales, ce qui nous permet de choisir les paramètres K_j , $j \in [1, 3]$, de façon à garantir la stabilité du système bouclé. La conception, intégrant des conditions suffisantes de stabilité, s'articule autour des étapes suivantes -l'annexe (2) détaille les différentes étapes de la méthode: réécriture du système bouclé, choix des domaines d'incertitude et résolution de l'inégalité matricielle:-

1. Les valeurs initiales des gains du PID sont calculées avec les formules de *Ziegler et Nichols* pour des échelons et sont modifiées pour les rampes -voir le paragraphe (3.2.1) du chapitre (3)-. La méthode du relais, appliquée sur des modèles discrets nominaux et sur une consigne moyenne, fournit les caractéristiques du point critique.
2. Le gain proportionnel est décomposé en deux termes $K_p \times K_p^*$. K_p permet de réaliser la compensation des variations du gain du système K_s , tel que $K_s \times K_p = K_f = 1$, la loi d'évolution de K_s étant estimée.
3. Le problème de stabilité posé concerne alors un système linéaire avec des paramètres incertains bornés $\delta_i(K_p, T_i, T_d)$, $i = 1$ à 3. Le système bouclé s'exprime selon l'équation $\dot{X} = \sum_i [\delta_i(K_p^*, T_i, T_d) A_i] X + A X$. Il est réécrit sous la forme

(5.7); les non linéarités sous forme d'incertitudes, $\delta_i(K_p^*, T_i, T_d)$, sont extraites de l'ensemble *PID supervisé + système*.

$$\begin{cases} \dot{X} = A X + B V \\ Y = C X \\ V = \delta(t) Y = \begin{pmatrix} \delta_1(t) & 0 & 0 \\ 0 & \delta_2(t) & 0 \\ 0 & 0 & \delta_3(t) \end{pmatrix} Y \end{cases} \quad (5.7)$$

4. Un choix initial des valeurs minimales et maximales des paramètres incertains est fait. Des conditions suffisantes de stabilité peuvent être déduites grâce à la découverte d'une fonction quadratique de *Lyapunov* appropriée, vérifiant l'équation (5.8) (établie à partir du critère général des petits gains structurés, [MAR,94]). R_i , Q_i et S_i caractérisent les incertitudes $\delta_i(t)$, P est la matrice de *Lyapunov* strictement positive cherchée, M et τ_i permettent de faciliter la recherche d'une fonction de *Lyapunov* de dérivée strictement négative.
5. Si la recherche est concluante, les valeurs de K_1 , K_2 et K_3 sont déduites de façon à ce que, pour la valeur maximale de α inférée (c'est-à-dire 1), K_p^* atteigne sa valeur maximale et T_i et T_d leur valeur minimale. Si la recherche n'aboutit pas, il faut recommencer à l'étape 3. L'annexe (2) fournit plus de précisions sur cette méthode, permettant de garantir la stabilité du système bouclé avec un PID supervisé.

$$\begin{pmatrix} X \\ V \end{pmatrix}^T \begin{bmatrix} A'P + PA + \epsilon I + \sum_i \tau_i R_i & PB + \text{col}(\tau_i S_i) \\ B'P + \text{col}[\tau_i(S_i)]' & \text{diag}(\tau_i Q_i) \end{bmatrix} \begin{pmatrix} X \\ V \end{pmatrix} \leq 0 \quad (5.8)$$

5.2.3 Choix effectués pour les paramètres du superviseur et Indications sur les liens avec les performances

A- Ensembles flous

Il est nécessaire de définir, tout d'abord, le nombre, la forme et la répartition des ensembles flous sur les différents univers de discours. Le nombre d'ensembles flous a été fixé à 9 pour e et de et 11 pour α . Ce choix repose sur des optimisations expérimentales à partir d'un nombre initial de 17 EF. Les ensembles flous considérés sont : *NG* (*Négatif Grand*), *NM* (*Négatif Moyen*), *NP* (*Négatif Petit*), *NTP* (*Négatif Très Petit*), *ZE* (*Zéro*), *PTP* (*Positif Très Petit*), *PP* (*Positif Petit*), *PM* (*Positif Moyen*), et *PG* (*Positif Grand*), auxquels se rajoutent *NTG* et *PTG* pour α .

Leur forme a été choisie triangulaire, pour simplifier les traitements informatiques, sachant que la forme a peu d'influence sur les résultats. La répartition des ensembles flous sur les univers de discours de e et de a été choisie non linéaire avec une concentration autour de l'origine, voir la figure (5.3). Cette distribution permet de réaliser des réglages du PID plus fins près du régime permanent.

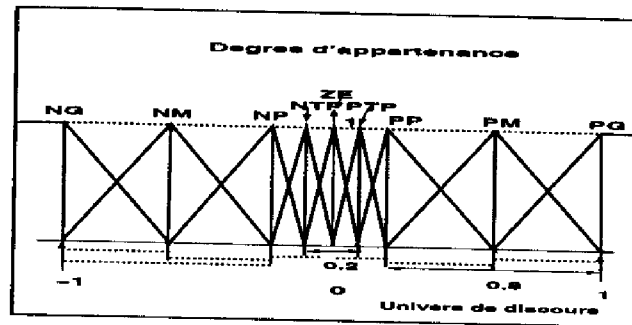


Figure 5.3 : Ensembles flous pour les différentes variables du superviseur flou.

B- Bornes des univers de discours

Les signaux erreur de sortie du four, e , et dérivée de cette erreur, de , sont normalisés dans l'intervalle $[-1, 1]$ au sein du module de normalisation. Les choix des bornes des univers de discours, e_{max} et de_{max} , sont faits de façon à utiliser au mieux la table de règles pour chaque changement de consigne en échelon ou en rampe. L'objectif est, évidemment, d'obtenir de bonnes performances; malheureusement, il est difficile de lier ces paramètres à un modèle de référence explicite de bonne convergence. Toutefois, quelques indications grossières peuvent être proposées.

Pour les suivis d'échelons, e_{max} , exprimé par un pourcentage de la variation de niveau de consigne -équation (5.9) et figure (5.4)-, peut traduire le dépassement maximal tolérable à chaque transitoire, et être lié à l'étendue de la bande proportionnelle.

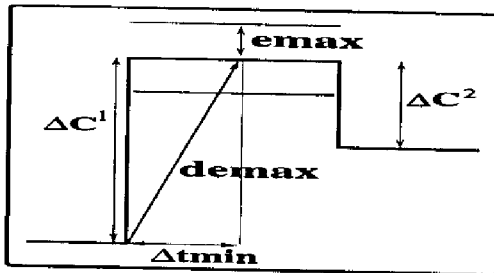


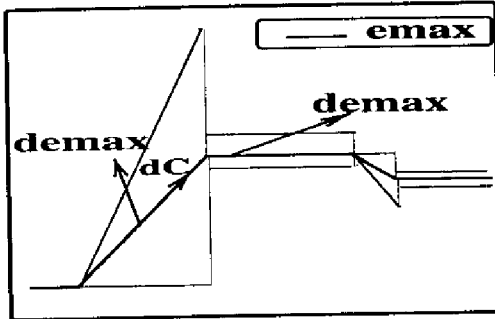
Figure 5.4 : Choix de e_{max} et de_{max} pour les échelons.

$$e_{max} = \frac{f_e}{100} \times \Delta C \quad (5.9)$$

$$de_{max} = dy_{max} = \frac{\Delta C}{\Delta t_{min}} \quad (5.10)$$

de_{max} peut être relié au temps de montée à 100% -équation(5.10)-. En effet, une diminution de Δt_{min} (performance attendue plus contraignante) entraîne une augmentation de de_{max} . Les ensembles flous, exprimant les valeurs d'une entrée de donnée, vont alors être plus proches de ZE. Si nous considérons, par exemple, une phase de convergence avec $e : PG$ et $de : N$, les tables de règles (5.1) ou (5.2) fournissent alors une valeur floue de α moins négative, entraînant une augmentation de la dynamique du régulateur. Ce changement produit une augmentation de la pente de la sortie du système et correspond à un temps de réponse plus rapide.

Pour les **suis de rampe**, e_{max} est variable lors du suivi de rampe et constant lorsque la référence est constante -équations (5.11) et (5.12) et figure (5.5)-.



$$e_{max} = pente \times t_{relatif} \quad (5.11)$$

$$e_{max} = \frac{f_e}{100} \times C \quad (5.12)$$

Figure 5.5 : Choix de e_{max} et de_{max} pour les rampes.

$$de_{max} = dy_{max} - dC = 4 \frac{f_{de}}{100} \times dC \quad (5.13)$$

Dans le cas de la formule (5.11), l'idée est d'introduire une stratégie spécifique d'optimisation du suivi de rampe; *pente* est la pente de la rampe et $t_{relatif}$ est le temps actuel, réinitialisé à 0 en début de rampe. e_{max} dépend de la distance au début de la rampe de telle façon que:

- au début du suivi de rampe, la réponse du four tend à diverger. L'objectif est alors d'anticiper sur cette divergence afin de forcer la réponse à reconverger le plus vite possible vers la rampe. Ce but est atteint en définissant un univers de discours petit. Ainsi, une saturation est introduite très rapidement et une action forte sera générée pour faire réagir le four plus vite et plus fortement.
- dans la suite du suivi de rampe, la réponse converge vers la référence, l'idée est alors d'atténuer les actions menées afin d'induire une convergence assez progressive pour éviter les dépassements. Les actions du superviseur sont alors progressivement diminuées par l'élargissement de l'univers de discours de e .

Dans le cas de la formule (5.12), il est possible de relier e_{max} au dépassement maximal accepté, en définissant un domaine autour de la consigne.

Nous pouvons donner des indications pour le choix de de_{max} dans le cas de la stratégie de supervision 1 proposée -formule (5.13)-. Ce paramètre peut être lié à la pente de la réponse du système attendue près de la référence; un critère de choix est, par exemple, une réponse peu oscillatoire. Lors du suivi de rampe, il est possible d'exprimer la pente de la réponse dy en fonction de la pente de la rampe dC ; l'équation (5.13) propose une expression de de_{max} . Le facteur 4 traduit approximativement le rapport entre une petite dérivée de l'erreur de^P et la valeur maximale de^G , sachant qu'il y a au total 9 ensembles flous pour coder les valeurs de de .

L'exemple suivant permet d'illustrer le fait que de^P est la dérivée de l'erreur servant de référence près de la consigne, dans le cas de la table de *Mac Vicar-Welhan*.

Sur la figure (5.6), la sortie du four est proche de la consigne (erreur petite et positive) et en phase de convergence vers la rampe. Deux cas de figure sont représentés pour une vitesse de convergence effective du signal de sortie égale, (cas 3), et plus grande, (cas 4), en valeur absolue que dY^P . Dans le cas (3), une action nulle est exercée car la sortie du four converge à la vitesse désirée vers la rampe, voir la table de règles sur la même figure. Dans le cas (4), une action anticipatrice importante, (NM), est imposée, pour minimiser les dépassements à venir. Le choix de de^P est donc lié à un comportement près de la consigne plus ou moins oscillatoire.

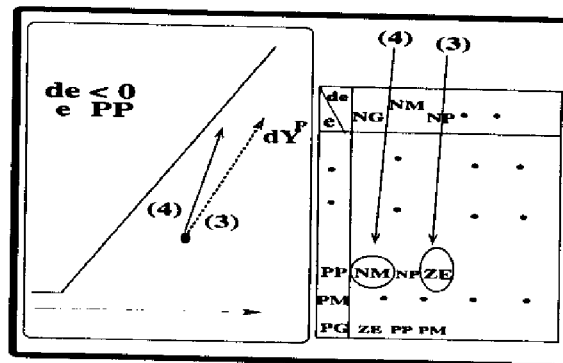


Figure 5.6 : Actions inférées en fonction des valeurs relatives de de et de_{max} .

5.3 Applications sur les fours

5.3.1 Application de la première stratégie

A- Simulations

Dans un premier temps, est utilisé un système du second ordre avec gain variable -selon la loi parabolique explicitée dans le paragraphe (1.4.2) du chapitre (1)-, caractérisé par l'équation (5.14). Dans un deuxième temps, est ajouté un retard pur non modélisé, voir l'équation (5.15). La constante d'échantillonnage est de 10s.

$$\frac{K_s}{(1 + 200p)^2} \quad (5.14)$$

$$\frac{K_s \times e^{-80p}}{(1 + 200p)^2} \quad (5.15)$$

Les différentes simulations ont été réalisées avec des séquences de plusieurs rampes, afin d'évaluer globalement le comportement du régulateur en présence de niveaux et de changements de consignes variés, [JOC,93]. Les évaluations chiffrées des performances atteintes sont le temps de stabilisation à 5% - T_{stab} - et le pourcentage de dépassement maximal par rapport au point de régulation - D_{max} -.

Pour le premier système, $K_1 = \frac{K_{p_{initial}}}{12}$, $K_2 = \frac{T_{i_{initial}}}{2}$ et $K_3 = \frac{T_{d_{initial}}}{3}$ permettent d'obtenir des performances intéressantes. Pour le second système, avec un retard pur important, une stratégie de supervision spécifique a été définie pour renforcer la stabilité du four, en minimisant davantage les oscillations. Pour cela, $K_1 = \frac{K_{p_{initial}}}{80}$, $K_2 = \frac{T_{i_{initial}}}{4}$ et $K_3 = 4 \frac{T_{d_{initial}}}{3}$.

Pour le **modèle du second ordre avec perturbations sur le gain**, les résultats donnés sur la figure (5.7), comparent les cas supervisé et non supervisé. La moyenne des améliorations sont de 7mn pour le temps de stabilisation et de 13.6% pour le pourcentage de dépassement. De plus, le suivi de rampe est également généralement amélioré. Le superviseur flou apporte une amélioration significative des performances en boucle fermée.

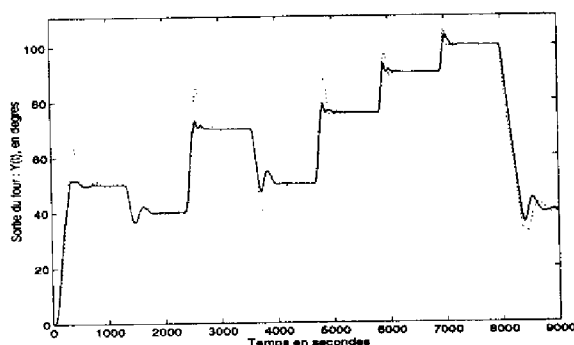


Figure 5.7 : Sortie du four avec gain variable: PID (..), PID supervisé (-).

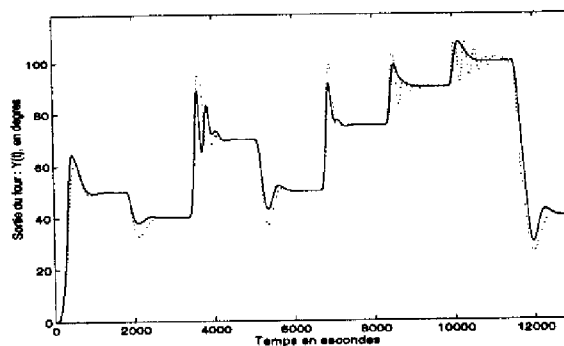


Figure 5.8 : Retard maximal non modélisé: PID (..), PID supervisé (-).

Pour le **modèle du second ordre avec perturbations sur le gain et retard pur**, la courbe (5.8) montre les résultats obtenus. Dans ce cas très contraignant, les moyennes des améliorations sont de 4mn pour le temps de stabilisation et de 10% pour le pourcentage de dépassement. Les amplitudes des premiers dépassements sont globalement minimisées et la fréquence des oscillations est réduite; ce phénomène s'accompagne, par ailleurs, d'un amortissement plus important de la réponse du four.

Les résultats des simulations montrent que, dans tous les cas, les régulations PID académiques sont améliorées grâce au superviseur flou, que ce soit au niveau de la poursuite de trajectoire ou au niveau de la stabilisation autour du niveau de référence constant.

B- Réglette à infra rouge

L'objectif des expérimentations temps réel sur la réglette à infra rouge -voir le paragraphe (1.4.2) du chapitre (1)- est de comparer les résultats de la régulation PID+CAS et PID+CAS supervisée pour montrer concrètement ce que la supervision floue peut apporter. La constante d'échantillonnage utilisée pour le superviseur est de 5s et le

régulateur communique avec le four toutes les 1s. Le régulateur PID+CAS est réglé initialement par un expert industriel. Des résultats sont obtenus pour une rampe et un échelon.

Pour le suivi de rampe avec un plateau à 50°, les valeurs des gains sont respectivement: $X_p = 20$, $T_i = 0$, $T_d = 27$ et $CAS = 75$. L'action intégrale étant gelée pour la régulation non supervisée, il est nécessaire de définir le facteur K_2 comme une plage de variation autour d'une valeur non nulle de l'action intégrale. En effet, le fait de garder comme valeur initiale de T_i la valeur 0, lors du régime de référence constante, conduit à des sauts de commande, donc à des oscillations. La valeur autour de laquelle s'effectue les variations de la constante de temps intégrale, lors de la consigne constante, a été choisie égale à 126s; la plage de variation autour de cette valeur médiane est de ± 120 s. Les paramètres K_1 , K_3 et K_4 ont été fixés respectivement aux valeurs 20, 10 et 0.05.

Pour la régulation autour d'un échelon à 100°C: $X_p = 40$, $T_i = 10$ mn, $T_d = 8$ mn et $CAS = 50\%$. Les gains du superviseur ont été réglés avec les valeurs: $f_{de} = 15$ et $f_e = 0.1$. Les paramètres K_1 , K_2 , K_3 et K_4 ont été fixés respectivement aux valeurs 20, 15, 10 et 0.05.

Pour la consigne en rampe, les résultats sont présentés sur les courbes (5.9) (réponses du four) et (5.10) (sortie du superviseur).

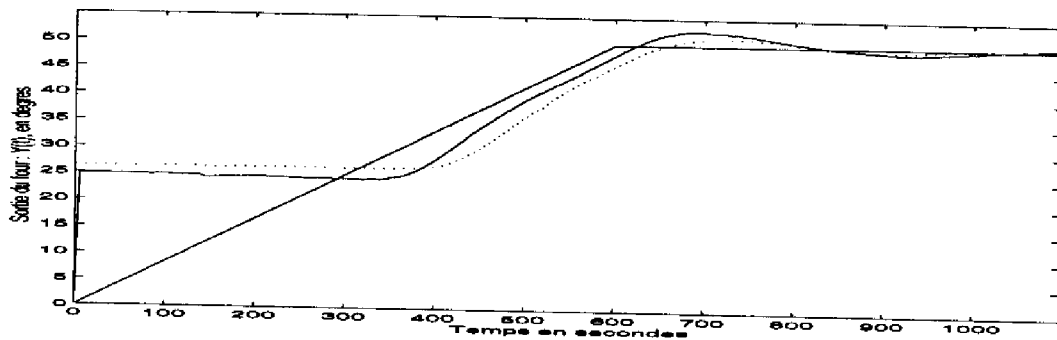


Figure 5.9 : Sortie de la réglette à infra rouge: avec le SF (-), sans le SF (..).

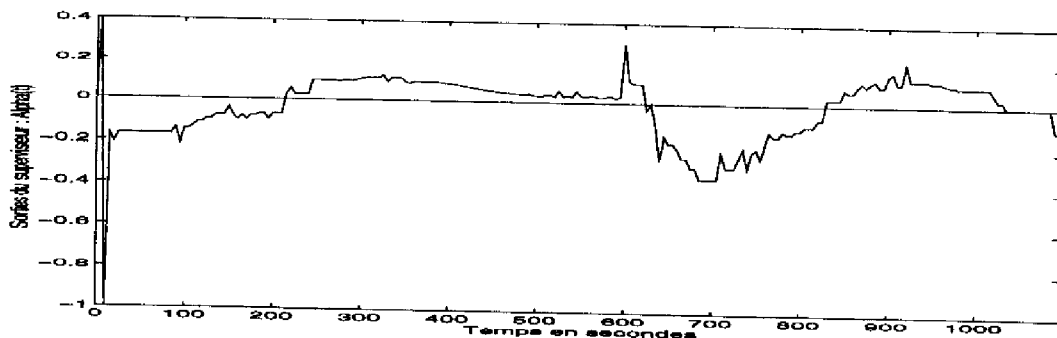


Figure 5.10 : Sortie du superviseur flou de PID.

Le suivi de rampe est nettement meilleur avec le PID+CAS supervisé qu'avec le PID+CAS non supervisé; l'erreur de trainage en fin de suivi de rampe passe de 4° à 1.6° . Le four réagit plus vite avec le superviseur, puisque, pour un premier dépassement plus important (il passe de 1.8° à 2.7°), le deuxième instant d'intersection avec le plateau est sensiblement le même (environ 237s après le début du plateau). La courbe (5.10) montre que, pendant le suivi de rampe, des actions positives de supervision sont menées afin d'augmenter la dynamique du régulateur; arrivé au plateau, des actions négatives permettent de rendre le régulateur moins rapide pour diminuer les oscillations.

Pour la **consigne en échelon**, les résultats sont présentés sur les courbes (5.11) (réponses du four) et (5.12) (sortie du superviseur). Pour un faible dépassement (il est de 0° pour le PID+CAS et de 0.8° pour le régulateur supervisé), le temps de montée à 100% est intéressant puisqu'il est de 3.4mn (au lieu de 4.8mn pour le régulateur non adapté en ligne). La courbe (5.12) montre que ceci est la conséquence des actions de supervision positives menées pendant le régime transitoire. Pendant le régime permanent, des actions négatives tendent à anticiper davantage sur les dépassements.

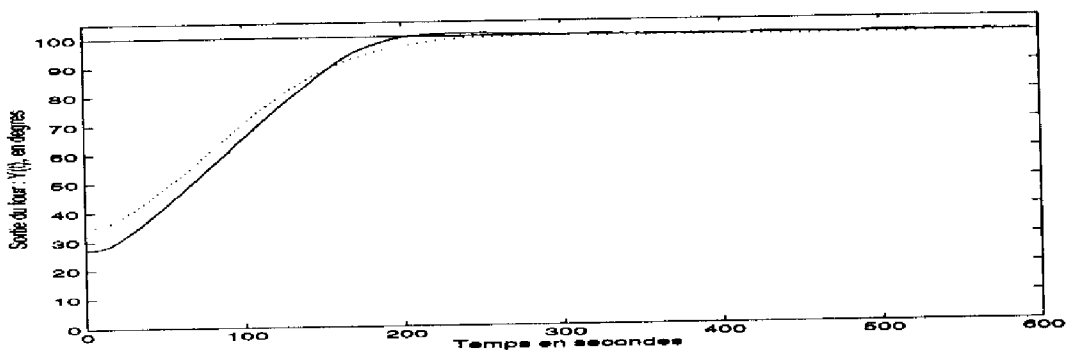


Figure 5.11 : Sortie de la rélette à infra rouge: avec le SF (-), sans le SF (..).

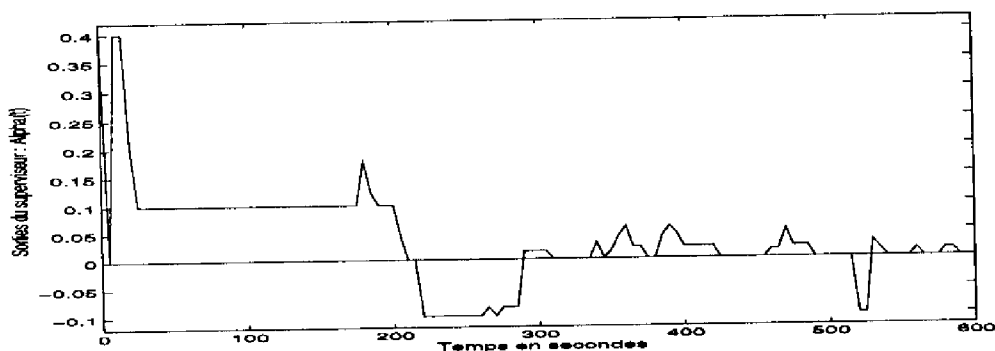


Figure 5.12 : Sortie du superviseur flou de PID.

5.3.2 Application de la deuxième stratégie en simulation

Des tests ont été menés avec le modèle (5.14) sur des successions de 10 transitoires en rampe ou en échelon. Les performances moyennes suivantes ont été évaluées: premier dépassement $-D_{max}$, temps de stabilisation à 5% $-t_{stab}$, allure du suivi de rampe (pour les rampes) et premier dépassement $-D_{max}$, temps de montée à 100% $-t_m$ et temps de réponse à 5% $-t_r$ (pour les échelons).

A- Références en rampe

La figure (5.13) montre un détail des deux premiers transitoires. La figure (5.14) focalise sur les actions de supervision et les évolutions des gains du PID.

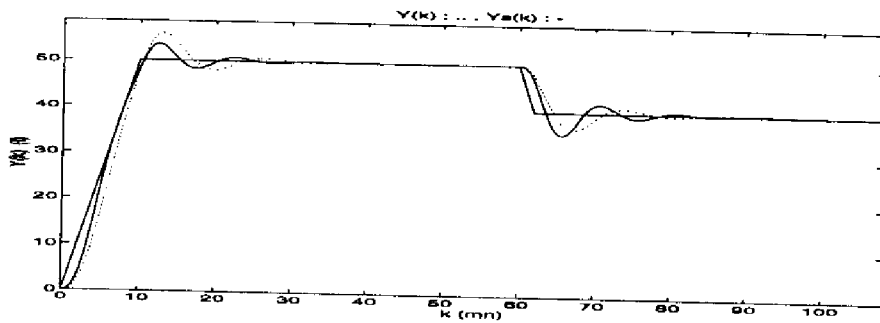


Figure 5.13 : Sortie du four: avec le SF (-), sans le SF (..) (rampes).

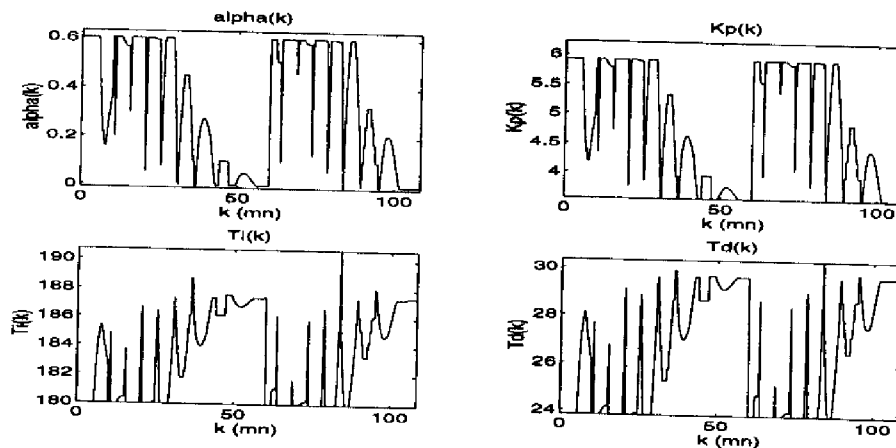


Figure 5.14 : Sortie du superviseur et gains du PID (rampes).

La sortie du superviseur α est toujours positive afin d'accélérer le suivi de rampe ou la stabilisation autour du plateau. Ceci se traduit, pour le superviseur, par une amélioration des trois indices de performance pour presque tous les régimes transitoires. Le tableau (5.3) résume les moyennes de ces améliorations.

	PID	PID supervisé	améliorations (%)
D_{max} (%)	13.4	11.74	+12.4
T_{stab} (mn)	10.57	9.3	+12
<i>suivi de rampe</i>	-	-	PG

Tableau 5.3 : Performances dynamiques moyennes et améliorations (rampes).

B- Références en échelon

La même analyse que précédemment est faite, voir les figures (5.15) et (5.16).

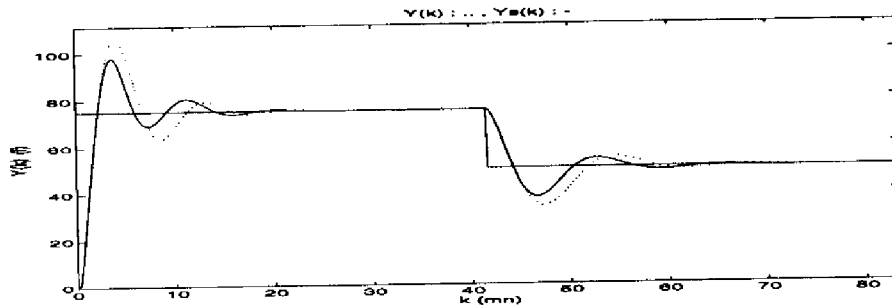


Figure 5.15 : Sortie du four: avec le SF (-), sans le SF (..) (échelons).

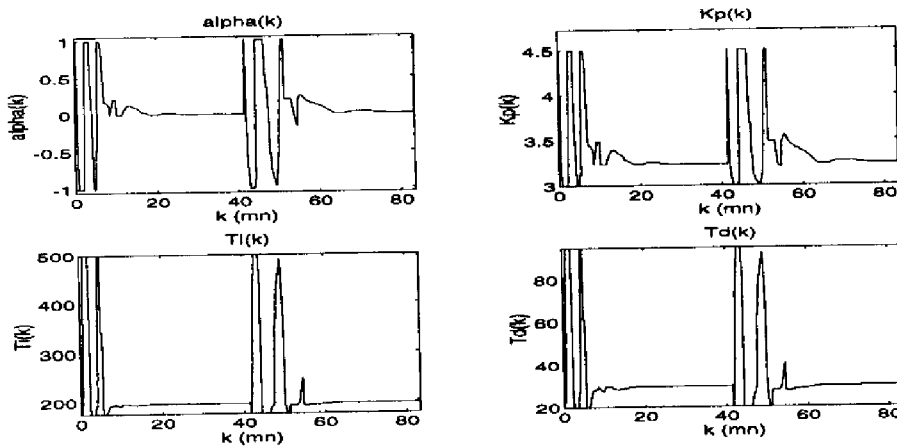


Figure 5.16 : Sortie du superviseur et gains du PID (échelons).

Les actions de supervision sont positives au début des transitoires et en phase de divergence au-delà de la consigne; elles sont négatives en dessous, pour anticiper sur les dépassements. Les améliorations de toutes les performances dynamiques pour tous les transitoires sont résumées dans le tableau (5.4).

	PID	PID supervisé	améliorations (%)
D_{max} (%)	28.5	20.7	+27
T_m (mn)	2.46	2.41	+2
T_r (mn)	11.5	8.6	+25

Tableau 5.4 : Performances dynamiques moyennes et améliorations (échelons).

5.4 Conclusions et extensions

A- Conclusions sur les résultats obtenus

La supervision floue de régulateurs PID est une façon intéressante de générer des lois non linéaires. Les principaux avantages de cette démarche sont:

1. une connaissance importante de l'organe de régulation PID dans le milieu industriel;
2. une approche floue qui permet de raisonner dans le plan de phase linguistique pour établir la base de règles;
3. une plus grande facilité d'utilisation des trois gains P, I et D que pour les régulateurs flous.

Le choix d'une structure de superviseur flou et les réglages de ses paramètres ont été explicités dans le paragraphe (5.2). Une seule table de règles de supervision exprime les valeurs floues d'un paramètre α , en fonction de l'erreur de sortie du four et de sa dérivée. Les gains du régulateur sont alors déduits de ce paramètre α . La table de règles et les formules d'adaptation permettent de fixer la logique de supervision caractérisant les liens entre paramètres du PID et performances en boucle fermée. Nous avons proposé deux approches de réglage du superviseur flou. La deuxième stratégie de supervision proposée permet, à travers le choix des coefficients K_j , d'assurer la stabilité du système bouclé.

Les simulations et tests sur le simulateur à infra rouge ont été présentés dans le paragraphe (5.3). Dans tous les cas, les résultats obtenus sont intéressants, puisqu'ils consistent en des améliorations de toutes les régulations PID académiques ou industrielles.

B- Extensions possibles du travail

Un des principaux inconvénients des méthodes de supervision floue proposées est la difficulté de définir un modèle de référence explicite pour les performances attendues. Il est recommandé de combiner les deux approches de supervision floue (adaptation en fonction de la position dans le plan de phase et en fonction des performances dynamiques atteintes à l'issue d'un régime transitoire). La première approche permet une optimisation constante de la loi de régulation et l'obtention de garanties de stabilité; la deuxième approche offre la possibilité à l'utilisateur de définir clairement les spécifications sur les performances. Il serait, en effet, intéressant de développer une interface homme-machine du superviseur, qui permette à l'utilisateur de définir, au niveau logiciel, le signal de référence, de spécifier numériquement et précisément ses exigences de performances en boucle fermée....

Enfin, d'autres modules pourraient être ajoutés au superviseur afin de compléter son fonctionnement. Par exemple, un module de planification de gains (*gain scheduling*) permettrait d'enregistrer des réglages pour des contextes donnés, dans des situations types, afin de restituer la configuration le cas échéant pour aller plus vite vers un réglage optimal. Dans le cas de systèmes avec une non linéarité sur le gain difficilement compensable, une technique de planification de gains, couplée à l'adaptation floue du PID autour des gains correspondants à une consigne donnée, permettrait de résoudre efficacement le problème de régulation.

Nous allons maintenant considérer une nouvelle façon d'améliorer des résultats de régulation PI avec seulement des techniques floues.

Chapitre 6

Auto-réglage de régulateurs flous

Nous proposons ici une autre approche pour améliorer une loi de régulation initiale de type PI, en travaillant exclusivement avec des techniques floues. Le RF de la deuxième génération développé consiste en un auto-régleur flou d'un système flou.

Nous avons amélioré une technique d'adaptation de singletons-conclusions de règles de régulation floue. En effet, nous proposons la définition d'une zone de non-adaptation des règles, en dessous de certaines performances limites, et une adaptation bidirectionnelle des singletons sur leur universs de discours. Nous avons également établi une formule d'adaptation de la variation de puissance, pour obtenir une bonne régulation dans le cas de systèmes des retards non modélisés importants.

Le système de commande a été validé en simulation et en temps réel, par analyse des lois de régulation obtenues et comparaison avec le RF de la première génération et avec un PID industriel.

Le paragraphe (6.1) donne des indications sur les méthodes de génération de lois floues non linéaires. Les paragraphes (6.2) et (6.3) décrivent les auto-régulateurs mis en oeuvre avec adaptations mono et bi-directionnelles sur l'univers de discours de la commande. En (6.4) sont présentées les applications sur les fours et en (6.5) les conclusions.

6.1 Génération automatique de lois de régulation floue non linéaire

6.1.1 Justification d'une telle démarche

La nécessité de développer des procédures d'auto-réglage est clairement apparue dans le paragraphe (3.2.2) du chapitre (3). Les régulateurs flous sont principalement intéressants pour le formalisme linguistique, permettant d'exprimer la stratégie de régulation, et pour leur aptitude à générer des lois de régulation non linéaires, compte tenu de leurs nombreux degrés de liberté -voir le paragraphe (1.2.2) du chapitre (1)-. Toutefois, ces nombreux paramètres ne sont pas toujours faciles à régler optimalement. Le premier problème posé consiste donc à choisir un moyen d'introduire facilement des non-linéarités dans la loi de régulation.

Le problème de la correspondance, entre une loi de régulation numérique et un certain réglage du régulateur flou, peut se résoudre par l'une des voies suivantes:

- Tout d'abord, les lois de régulation non linéaires peuvent être conçues à partir de correspondances entre formalisme linguistique flou et fonctions analytiques. Toutes les équivalences ne sont pas connues; mais on sait qu'un système à base de règles peut approximer n'importe quelle non-linéarité, au même titre qu'un réseau de neurones. La connaissance des équivalences peut faciliter des études de stabilité. Toutefois, un des attraits essentiels du flou, à savoir la modélisation de connaissance experte, peut disparaître.
- C'est pourquoi, il est intéressant d'envisager l'adaptation des paramètres du RF en utilisant des techniques d'optimisation. Ces méthodes reviennent à construire ou modifier la loi de régulation, à partir de données expérimentales. Deux niveaux sont possibles pour ces réglages. Un auto-calibrage des paramètres du régulateur flou peut être réalisé a priori, en fonction des caractéristiques du système et du signal de consigne, comme la méthode du relais avec le PID. Une auto-adaptation des paramètres du régulateur flou peut également être menée en ligne, pour réagir aux perturbations internes, externes ou aux changements de niveaux de consigne. Ces méthodes adaptatives permettent de régler automatiquement l'organe de commande à partir de spécifications sur les performances dynamiques en boucle fermée. En contrepartie, la démarche suivie conduit malheureusement à une impossibilité d'analyse de stabilité a priori.

Nous avons opté pour une combinaison des deux approches: une loi de régulation linéaire connue est d'abord traduite en flou, puis modifiée localement pour introduire des non-linéarités. Bien que partant d'une loi initiale linéaire, un des avantages du flou, qui est l'intégration de la connaissance experte, n'est pas ignoré ici. En effet, lors de l'étape d'introduction de non-linéarités locales, cette connaissance peut s'avérer utile. Cette démarche conduit à une garantie d'amélioration des performances par

rapport à une régulation linéaire et à plus de présomptions de maintien de la stabilité, que dans le cas d'un régulateur flou réglé de façon complètement heuristique. Un exemple de procédure d'auto-réglage pour les régulateurs flous est donné dans [HAY,91]. La technique repose sur une adaptation du régulateur en fonction de performances dynamiques observées à l'issue d'un transitoire. Nous nous sommes inspirée de cette application.

6.1.2 Sur quels paramètres du RF agir et comment ?

Dans le paragraphe (2.2) du chapitre (2), nous avons déjà évoqué les paramètres principaux sur lesquels il est intéressant d'agir pour introduire des non-linéarités dans la loi de régulation. Il est ainsi possible de modifier la nature des interpolations générées pendant le processus d'inférence, ou directement les amplitudes des niveaux de commande:

- Dans le premier cas, modifier les interpolations naturelles entre les points caractéristiques peut se faire en agissant sur la représentation des ensembles flous des variables d'entrée (forme et répartition des EF -incluant les recouvrements entre deux EF consécutifs et les positions des valeurs modales-), ou sur le mécanisme d'inférence (donc sur les opérateurs mathématiques utilisés). Ce cas d'introduction de non-linéarités est intéressant si peu de règles sont définies.
- Dans la situation contraire, il est possible d'introduire des non-linéarités par une démarche heuristique experte (ex.: utilisation d'une méthode qualitative pour modifier localement les règles dans le plan de phase linguistique), ou par des techniques d'auto-réglage.

Nous avons déjà montré dans le paragraphe (3.2.2) du chapitre (3) que *certaines interpolations non linéaires peuvent améliorer les résultats. Nous nous intéressons maintenant à la modification automatique des règles.* Il est intéressant, pour cela, d'agir directement sur les conclusions des règles. Le fait de conserver une couverture fixe de l'espace d'entrée permet de préserver une sémantique, même si tous les degrés de liberté permettant d'introduire des non-linéarités ne sont pas exploités. Nous allons montrer, dans la suite, que l'introduction de non-linéarités, par adaptation des conclusions des règles, permet d'améliorer notablement les performances dynamiques des régulateurs pour l'application qui nous concerne.

Lors de l'introduction automatique, de différentes façons, de non-linéarités, il est souvent question *d'apprentissage* même si ce terme doit être nuancé; aussi est-il préférable de parler *d'optimisation*:

- Des combinaisons flou/neuronal peuvent être mises en oeuvre, les réseaux de neurones étant appropriés pour le traitement de données expérimentales.

- D'autre part, des algorithmes génétiques peuvent être combinés aux modules flous. Ils permettent de générer des structures de régulation floue optimales; par exemple, ils permettent de combiner les avantages de deux structures, intéressantes pour des raisons différentes.
- Les techniques d'optimisation traditionnelles, appliquées sur le régulateur flou, permettent de générer des lois de régulation non linéaires (par exemple: descente de gradient, recuit simulé, simplexe....).
- Enfin, des techniques d'optimisation spécifiques peuvent être utilisées, telles des méthodes floues pour résoudre des problèmes éventuellement multi-critères.
Nous nous sommes intéressée, dans ce travail, à cette dernière approche.

Les types de non-linéarités introduites vont dépendre des critères d'auto-réglage utilisés. Les objectifs fixés pour améliorer la loi de commande peuvent concerner toutes sortes de performances dynamiques ou de combinaisons de ces performances (dépassements, temps de montée, temps de stabilisation, temps de réponse, facteur d'amortissement, erreur courante, puissance utilisée, somme cumulée des erreurs, intégrale de l'erreur quadratique....).

6.2 Mise en oeuvre d'un auto-régleur de RF: démarche générale

L'objectif de notre travail est de proposer une méthode d'auto-réglage permettant de générer des lois de régulation plus complexes que celles, quasi-linéaires, développée dans le paragraphe (3.2.2) du chapitre (3). Les lois non linéaires générées doivent permettre d'obtenir un meilleur compromis *dépassement-rapidité* qu'avec une loi linéaire et une meilleure robustesse face à des perturbations. Notre but est de développer un RF qui conduise à de meilleurs résultats qu'un PID industriel, qui soit facile, rapide à concevoir et à utiliser, et qui tienne compte des spécifications de l'utilisateur.

6.2.1 Principe général de l'auto-réglage

Les RF considérés possèdent une structure de base décrite dans le paragraphe (3.2.1) du chapitre (3). La méthodologie d'auto-réglage proposée est constituée de deux étapes.

Le réglage du régulateur est d'abord réalisé de façon à approximer une loi de régulation analytique de type PI rapide, d'après les lois d'équivalence exposées dans le paragraphe (3.2.1) du chapitre (3). Une première expérience de pré-réglage du type relais permet de fournir les gains proportionnel et intégral servant à initialiser les paramètres du régulateur flou.

A partir de la loi de régulation floue standard obtenue (notée RFB_0), l'auto-réglage est appliqué sur une séquence de consignes en rampe ou en échelon. Il consiste en une adaptation de la position des ensembles flous de la variable de sortie du régulateur, en fonction des performances dynamiques atteintes à la fin du transitoire précédent. Cette démarche conduit à la génération d'une loi de régulation floue non linéaire avec une base de règles adaptée (notée RFB_1).

6.2.2 Réglage d'un PI flou non linéaire

Le principe de l'auto-réglage est une modification des règles de convergence, afin de se rapprocher d'un modèle de référence implicite de la réponse temporelle. Les modifications consistent à moduler les dosages de ΔU en fonction des pourcentages de premier dépassement atteints, jusqu'à ce que ces dépassements soient suffisamment petits. Les règles concernées par les adaptations sont illustrées sur la figure (6.1). L'ensemble des règles caractérisé par un signe (+) correspond au cas où la sortie du système est au dessus de la consigne et est en train de converger vers cette référence. L'ensemble des règles caractérisé par un signe (-) correspond au cas où la sortie du four est en dessous de la consigne et est en train de converger vers cette référence.

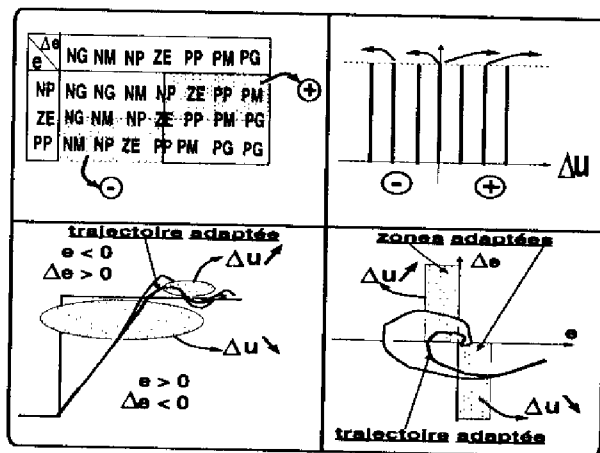
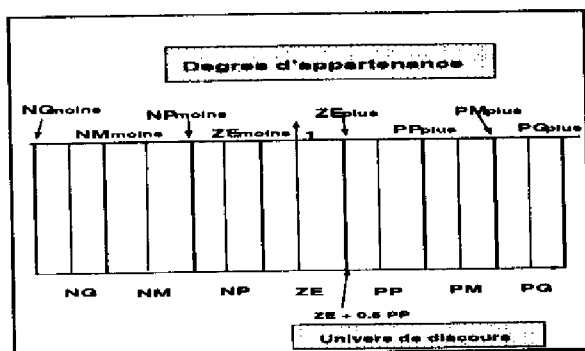


Figure 6.1 : Principe de l'adaptation des règles.

L'introduction de non-linéari-tés dans la loi de régulation floue est faite en déplaçant certains singletons flous de parties conclusions des règles, sur l'univers de discours de la variable *variation de commande*. L'idée de l'auto-réglage consiste à anticiper davantage sur les dépassements des réponses temporelles, liées à des dynamiques en boucle fermée rapides mais oscillatoires.

Ainsi, le sens d'adaptation de ces singletons va dans le sens d'un renforcement des actions anticipatrices. Les singletons flous positifs sont déplacés vers les valeurs croissantes sur l'univers de discours de ΔU , pour générer les nouveaux singletons ZE_{plus} , PP_{plus} , PM_{plus} et PG_{plus} . Les singletons négatifs sont déplacés vers les valeurs décroissantes sur l'univers de discours de ΔU , pour générer les nouveaux singletons NG_{moins} , NM_{moins} , NP_{moins} et ZE_{moins} . Ces mouvements se font, dans notre cas, dans des proportions égales pour tous les singletons adaptables. Le mécanisme de transformation d'ensembles flous en d'autres ensembles flous, pour la variation de commande, conduit donc à une modification du contenu de certaines règles.


 Figure 6.2 : Singleton optimisés pour ΔU .

La figure (6.2) illustre le résultat d'une phase d'adaptation de la base de règles, en montrant la définition des ensembles flous résultant pour ΔU . Les positions des nouveaux singletons sont définies avec un facteur de décalage par rapport aux singletons de départ; ici, il est de 0.5.

D'autres types de non-linéarités sous forme de saturations variables ont été introduites dans la loi de régulation, afin d'améliorer les résultats dans le cas de retards importants non modélisés. L'idée est d'adapter, au début de chaque nouvelle consigne, le facteur de variation de la puissance disponible, ΔU_{max} , tel que: si le gain du système K_s augmente, alors ΔU_{max} diminue et si le retard T_u augmente, alors ΔU_{max} diminue. Une loi analytique permet de traduire ces relations qualitatives; elle consiste donc en une adaptation de ΔU_{max} pour compenser implicitement les effets néfastes d'un retard pur, en plus des effets des perturbations internes sur le gain du four, voir l'équation (6.1). Cette adaptation s'avère nécessaire pour pouvoir mener à bien une régulation acceptable dans le cas d'un retard pur égal à 40% de la valeur de la constante de temps T_s .

Dans la formule considérée, $\Delta U_{maxinit}$ est la borne de l'univers de discours de ΔU , choisie au début de la simulation. K_{sinit} est le gain du four correspondant au premier plateau de référence, $K_{sactuel}$ est le gain actuel du four estimé pour le niveau de référence constant courant. $gain_{K_s}^{transitoire}$ sont les paramètres de pondération du gain réglé par essais successifs; $gain_{T_u}^{transitoire}$ sont les paramètres d'adaptation du retard pur (T_u est en nombre de périodes d'échantillonnage), réglés par essais successifs; l'indice *transitoire* est égal à (m) ou (d) pour caractériser un transitoire montant ou descendant.

$$\Delta U_{max} = gain_{K_s}^{transitoire} \Delta U_{maxinit} * \frac{K_{sinit}}{(K_{sactuel} + gain_{T_u}^{transitoire} T_u)} \quad (6.1)$$

6.2.3 Choix des paramètres par l'utilisateur

L'organe permettant d'adapter les règles est un module flou dont certains paramètres doivent être choisis par l'utilisateur. Le choix d'un système flou pour réaliser la tâche d'adaptation se justifie par le fait que les discontinuités dans les actions de réglage sont évitées, grâce au caractère interpolatif de l'inférence floue et à la représentation de connaissance graduelle avec les ensembles flous. De plus, la méthode de réglage

sera plus facile à concevoir et paramétrer car elle est plus globale, plus synthétique: un algorithme de type séquentiel répétitif nécessiterait d'énumérer exhaustivement les différents cas.

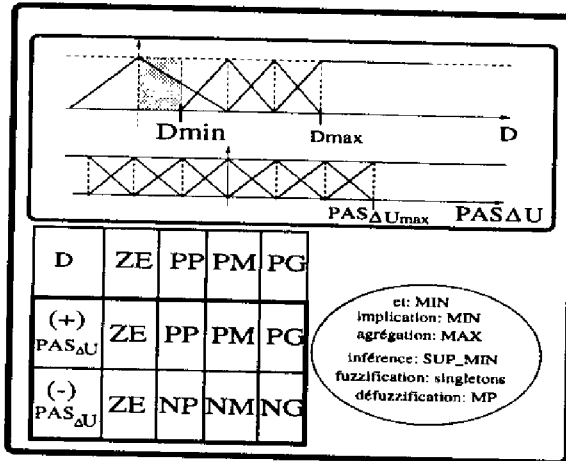


Figure 6.3 : Module d'adaptation des règles.

Le module flou d'adaptation des règles est donc composé d'un système flou à une entrée, le pourcentage de dépassement maximal du dernier transitoire par rapport au plateau courant (D), et une sortie, le pas de déplacement des singletons adaptables sur l'univers de discours de ΔU ($PAS_{\Delta U}$), voir la figure (6.3). Les deux variables sont caractérisées par des fonctions d'appartenance triangulaires.

Nous proposons de définir D sur un univers de discours où D_{min} définit la limite de la zone de tolérance pour les pourcentages de dépassement. En-dessous de D_{min} , aucune adaptation n'est menée, afin de ne pas adapter constamment les singletons et de ne pas converger vers une régulation trop amortie. D_{max} est défini automatiquement à partir de D_{min} . $PAS_{\Delta U}$ est définie sur un univers de discours où le paramètre à définir est $PAS_{\Delta U_{max}}$, la borne maximale. Les valeurs de déplacement des singletons sont interpolées à partir des règles de la figure (6.3) pour les règles de régulation floue (+). Pour les règles adaptables caractérisée par le signe (-), le pas d'adaptation est inversé.

L'utilisation de l'algorithme d'adaptation des règles du régulateur flou nécessite donc la définition des valeurs de D_{min} , dont le choix repose sur le compromis: *spécifications pas trop exigeantes / limitation intéressante des dépassements* et $PAS_{\Delta U_{max}}$, dont le choix repose sur la résolution du compromis *convergence rapide / comportement final pas trop amorti*. Ces paramètres sont choisis d'après les spécifications et l'expérience de l'utilisateur. Si les choix effectués conduisent à des réponses temporelles, en fin de phase d'auto-réglage, trop amorties ou encore trop oscillatoires, il est alors nécessaire de modifier les choix et de relancer la procédure d'auto-réglage. Par exemple, pour affermir les exigences sur les performances, il est souhaitable de diminuer D_{min} . D'autre part, une augmentation de $PAS_{\Delta U_{max}}$ correspond à une moins grande exigence sur la précision de convergence. La convergence vers un réglage optimal n'est donc pas garantie; la procédure n'est, à ce stade, pas complètement automatisée.

Si l'adaptation de la variation de puissance en fonction du gain et du retard du four est utilisée, il faut choisir de façon heuristique les quatre paramètres supplémentaires: $gain_{K_s}^m$, $gain_{K_s}^d$, $gain_{T_u}^m$ et $gain_{T_u}^d$, voir la formule (6.1).

6.3 Extension à une régulation adaptative: amélioration du module d'auto-réglage de base

6.3.1 Principe général

La procédure d'adaptation automatique des règles, présentée dans le paragraphe précédent, conduit à des résultats intéressants à condition de choisir astucieusement les deux paramètres de cette procédure, qui sont le pourcentage maximal de dépassement toléré et les pas d'adaptation maximal des singletons. En effet, la procédure ne permettant de déplacer les singletons que dans un sens sur les univers de discours, des exigences trop contraignantes en terme de dépassements ou un pas d'adaptation trop grand conduiraient à une réponse trop amortie et, donc, à la nécessité de relancer la procédure de réglage avec d'autres valeurs de ces deux paramètres.

Pour *apprendre* de façon autonome un comportement optimal défini par l'utilisateur, une solution consiste à modifier la procédure d'adaptation afin que la recherche du réglage optimal se fasse dans les deux sens sur l'univers de discours de ΔU , en fonction de critères tenant compte de la réalisation du compromis *réponse amortie/dépassement satisfaisant*. Ainsi le critère de performance fait intervenir le pourcentage de premier dépassement lié à un transitoire D , et T , qui représente le temps de montée à 100%. Le principe de base consiste à anticiper davantage sur les dépassements lorsque ceux-ci sont trop grands et, au contraire, à fournir plus de puissance si le comportement en boucle fermée est trop amorti. Une priorité est donnée à la minimisation des dépassements. La stratégie d'adaptation fait converger les performances dynamiques dans un domaine de valeurs souhaitées, donc vers un modèle de référence défini qualitativement par les ensembles flous ZE de D et T (voir la description du module d'adaptation dans le paragraphe suivant). La procédure d'adaptation de la base de règles, pour la méthode d'auto-réglage, est un bon départ pour une stratégie adaptative.

6.3.2 La table de règles

La table de règles utilisée, pour les règles de régulation floue (+), est présentée dans le tableau (6.1).

La position référence, pour laquelle aucune action n'est menée, est $(D, T) = (ZE, ZE)$. Les actions dans les autres régions de l'espace qualitatif (D, T) ont été déduites des recommandations suivantes. Dans la zone (1) (gras et souligné), T est acceptable, mais D est trop important; les actions visent alors à réduire D .

$T \downarrow D \rightarrow$	ZE	PP	PM	PG
ZE	ZE	PP	PM	PG
PP	NP	PP	PP	PM
PM	NM	NP	PP	PP
PG	NG	NM	NP	PP

Tableau 6.1 : Table de règles du module d'adaptation du RF.

Dans la zone (2) (gras), les dépassements sont trop grands, la priorité est donnée à leur réduction. Dans la zone (3) (souligné), D est faible, mais T devient trop important; la priorité est la réduction de T . Et enfin, dans la zone (4) (normal), T est trop grand devant D , il faut absolument réduire T .

6.3.3 Les choix des paramètres

La figure (6.4) représente les univers de discours des variables entrée-sortie du module d'adaptation flou.

Trois paramètres apparaissent sur ces représentations. Ce sont D_{min} , défini en pourcentage par rapport au plateau courant, T_{min} , exprimé en secondes et $PAS_{\Delta U_{max}}$, donné en pourcentage de ΔU_{max} . Les bornes des univers de discours des entrées du module d'adaptation, D_{max} et T_{max} , sont définies automatiquement, de façon à obtenir la distribution des ensembles flous de la figure (6.4). (D_{min}, T_{min}) définissent les performances dynamiques à ne pas dépasser, donc le modèle de référence analytique limite. Leur choix doit être fait judicieusement par l'utilisateur de manière à définir un compromis atteignable.

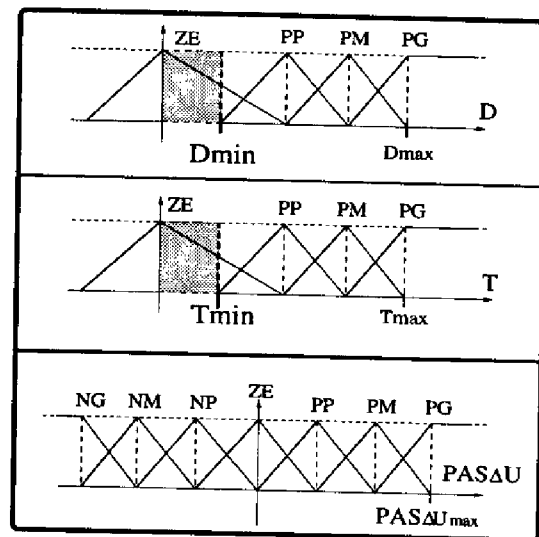


Figure 6.4 : Univers de discours des E-S du module d'adaptation.

$PAS_{\Delta U_{max}}$ fixe la vitesse de convergence. Une valeur trop grande conduit à une réponse oscillatoire et à une adaptation qui ne converge pas.

6.4 Application à la régulation de température de fours

Les applications ont été réalisées en simulation, sur des modèles de four du deuxième ordre avec perturbations internes, et en temps réel, sur le simulateur à réglette à infra rouge, voir le paragraphe (1.4.2) du chapitre (1).

6.4.1 Simulations d'auto-réglage de base

Les résultats présentés ici ont été obtenus sur le modèle nominal du four du second ordre avec un gain de 0.87, une constante de temps de 200s et une période d'échantillonnage de 10s, puis sur les modèles perturbés.

L'étude consiste d'abord en une analyse des propriétés des lois de régulation obtenues sur le modèle nominal. Dans ce cadre, est proposée une analyse des non-linéarités introduites, par comparaison entre les lois de régulation PI analytique, floue standard équivalente et floue adaptée.

Ensuite, les effets locaux des adaptations sont observés sur les réponses temporelles, afin de mettre en évidence les améliorations apportées sur les performances en boucle fermée. Pour mesurer l'adéquation des performances obtenues avec les spécifications du cahier des charges (faibles temps de montée, faibles dépassements en poursuite et régulation, pas d'erreur statique), sont évaluées: l'erreur en régime permanent, l'amplitude du premier dépassement $-D-$, le temps de montée à 100% $-T_m-$, le temps de stabilisation à 0.5% $-T_{stab}-$ et la qualité du suivi de rampe.

Enfin, une étude expérimentale comparative de la sensibilité des différents types de régulateurs, à des perturbations internes ou à des erreurs de modélisation, a été menée (aspect bonne réjection de perturbations du cahier des charges). L'évaluation de la sensibilité des régulateurs permet de commencer une estimation de la robustesse en performances. Le calcul des dégradations de performances sur un modèle perturbé par rapport au modèle nominal a été indiqué dans le paragraphe (2.3.1) du chapitre (2).

Les régulateurs flous utilisés sont ceux du paragraphe (3.2.1) du chapitre (3). Les phases d'auto-réglage des bases de règles des régulateurs flous standards conduisent à la définition des bases adaptées B_1 pour les rampes (facteur de décalage de 1.5) et B_2 pour les échelons (facteur de décalage 0.5).

A- Analyse de la loi de régulation obtenue

Les spécificités des non-linéarités, introduites par l'adaptation des règles de convergence, sont soulignées sur les courbes (6.5) à (6.7) (cas des rampes). La loi de régulation floue standard est remodelée. Les particularités de la loi floue adaptée sont illustrées sur les figures (6.8) et (6.9).

Dans la région du plan de phase correspondant à l'adaptation des règles (zones de convergence), les singletons utilisés sont les nouveaux ensembles flous translatés. La distance entre deux singletons consécutifs différents de *zéro* est constante; par contre, la distance de chaque singleton à l'origine augmente. Par conséquent, la pente selon la diagonale $(e, \Delta e) = (PP, NP) \Rightarrow (NP, PP)$ passe d'une valeur nulle à une valeur positive, liée à l'importance du facteur de décalage des singletons de ΔU .

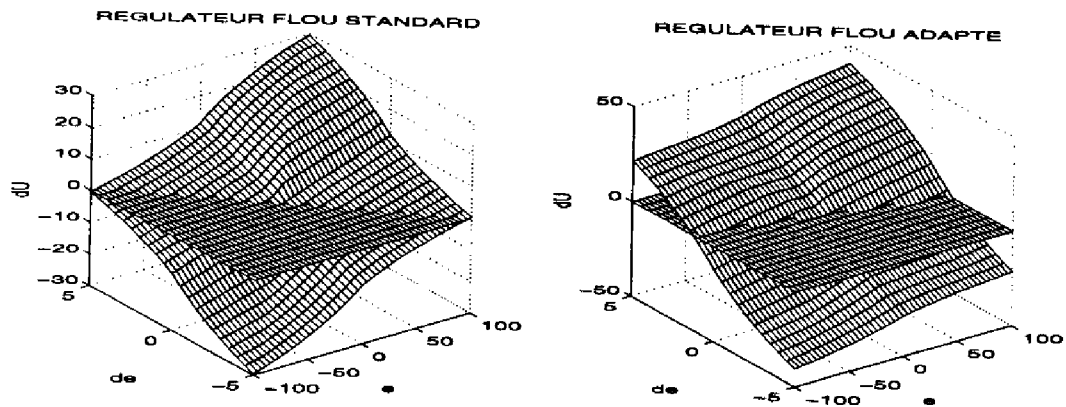


Figure 6.5 : Lois de régulation $\Delta U(e, \Delta e)$ avec les régulateurs flou standard et flou adapté.

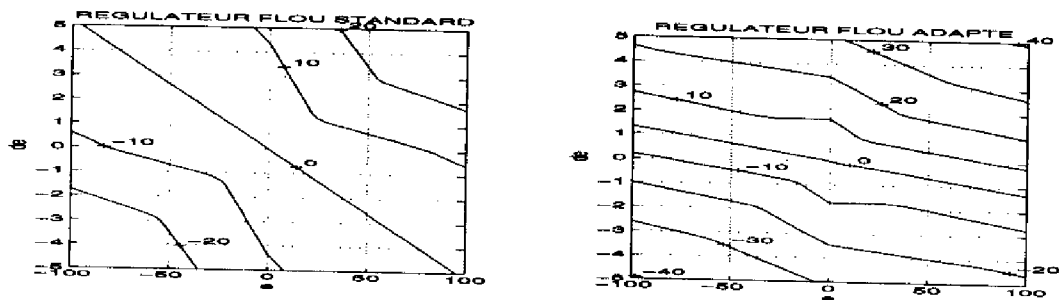


Figure 6.6 : Projections des lois de régulation $\Delta U(e, \Delta e)$ sur le plan de phase $e(\Delta e)$.

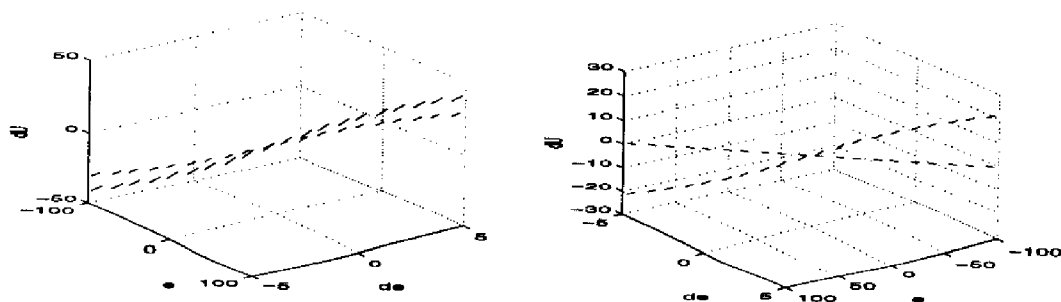


Figure 6.7 : Coupe des lois de régulation (régulateur flou adapté: courbes supérieures).

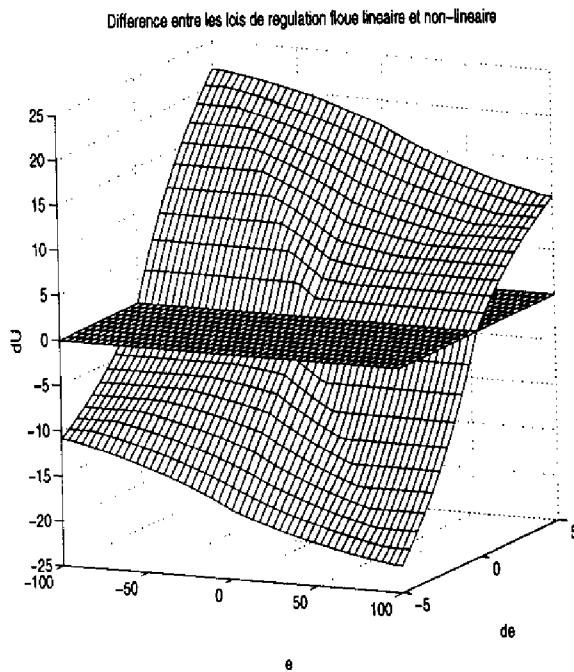


Figure 6.8 : Différences entre les lois floue adaptée et floue standard.

$e \backslash de$	NP	ZE	PP
NP	NM (d) -	NP	ZE (f) ++
ZE	NP (c) -	ZE	PG (a) +
PP	ZE	PP	PM

Figure 6.9 : Plan de phase linguistique: chaque zone linguistique a, b,..., est associée à l'importance de la différence entre les lois adaptée et standard (+, ++, ..), voir la figure ci-contre.

La pente selon la diagonale $(e, \Delta e) = (PP, PP) \Rightarrow (NP, NP)$, correspondant aux zones de divergence, est légèrement augmentée en raison de l'influence, durant le processus d'inférence, des règles voisines appartenant aux zones de convergence. Les lignes de niveaux de la figure (6.6) permettent de souligner les conséquences, sur la loi de régulation, des adaptations effectuées: l'accentuation des pentes selon les deux diagonales se retrouve dans le resserrement des lignes de niveaux et dans leur changement d'orientation. Les nouvelles cassures sont complètement différentes de celles de la loi de régulation floue standard. Les cassures dues à l'interpolation grossière ont disparu, au profit de changements de pentes dû aux modifications, dans des proportions différentes, de la loi de régulation pour les zones de convergence ou de divergence. Certaines influences néfastes du mécanisme d'interpolation utilisé dans le contexte d'une granularité floue très grossière sont maintenant complètement effacées -voir le paragraphe (3.2.2) du chapitre (3)-. De plus, le renforcement local des zones de convergence permet d'améliorer globalement la réponse dynamique du four, comme on va le voir ci-dessous.

B- Performances dynamiques obtenues

Les courbes (6.10) et (6.11) présentent les résultats comparatifs des poursuites et régulations.

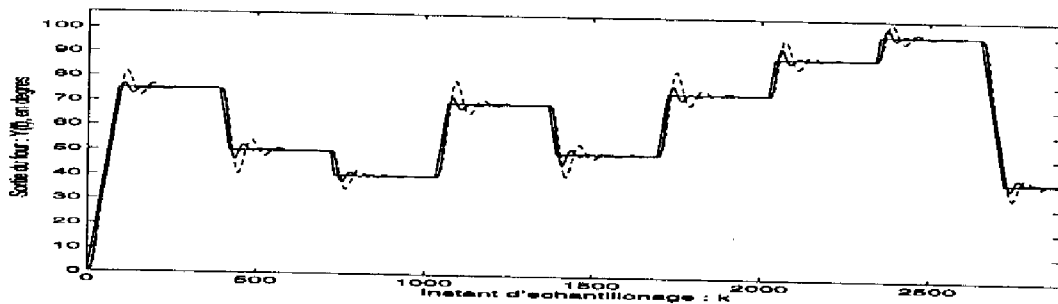


Figure 6.10 : Réponses du four nominal: RF standard (- -), RF adapté (-).

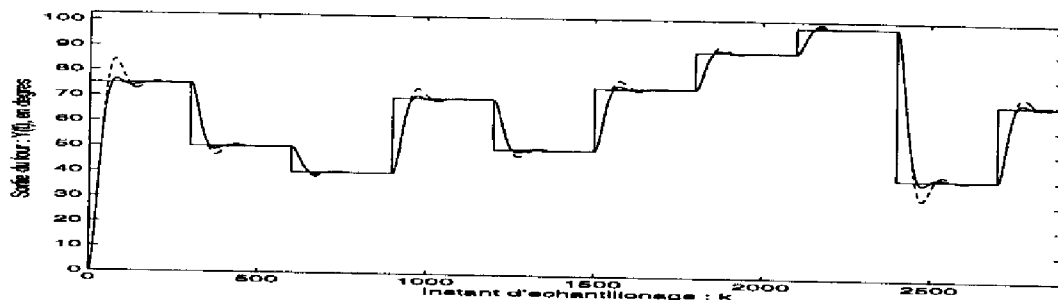


Figure 6.11 : Réponses du four nominal: RF standard (- -), RF adapté (-).

Dans les deux cas, les résultats obtenus sont globalement meilleurs qu'avec les régulateurs PI ou flou standard (noté RFB_0). Les tableaux (6.2) et (6.3) résument les performances moyennes obtenues pour chaque type de régulateur. Avec le RF adapté (noté RFB_1), les premiers dépassements et les comportements oscillatoires sont fortement atténués. Dans le cas de références en échelon, la rapidité de la réponse est légèrement meilleure en début de régime transitoire, bien que le temps de montée moyen soit un peu plus élevé. Le suivi des rampes est plus intéressant.

	D (%)	T_{stab} (mn)	suivi de rampe
PI	9.7	26.8	Moyen
RFB_0	12.7	45.7	Bon
RFB_1	5	17	Très Bon

Tableau 6.2 : Performances (rampes).

	D (%)	T_m (mn)
PI	7.2	8.4
RFB_0	6.5	8.2
RFB_1	1.7	8.9

Tableau 6.3 : Performances (échelons).

Malgré l'utilisation d'une zone réduite de la table de règles et d'un mécanisme d'interpolation très grossier -voir le paragraphe (3.2.2) du chapitre(3)-, l'adaptation simple des zones de convergence conduit à un meilleur dosage des commandes. Dans tous les cas, le régulateur flou adapté agit plus efficacement, car il réagit plus tôt et plus vivement. L'adaptation des zones de convergence produit un renforcement des actions d'anticipation sur les dépassements. Toutefois, une trop forte adaptation des singletons des zones de convergence peut conduire à des réponses trop amorties; cette limite réduit les possibilités d'atténuation des dépassements.

C- Analyse de sensibilité

a- Variations du gain du four: La première perturbation appliquée est la modification du gain du four K_s selon la loi (1.10) du chapitre (1). Les résultats sont présentés sur les figures: (6.12), (6.14) (suivis de rampes) et (6.13), (6.14) (régulations autour d'échelons).

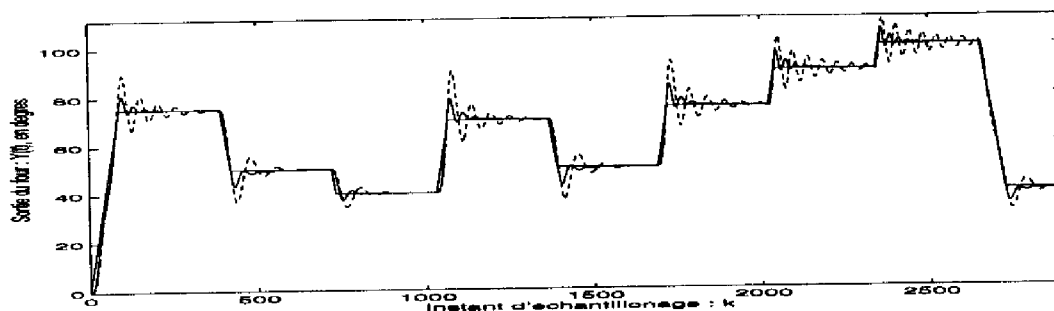


Figure 6.12 : Gain variable: régulateur flou standard (- -), régulateur flou adapté (-).

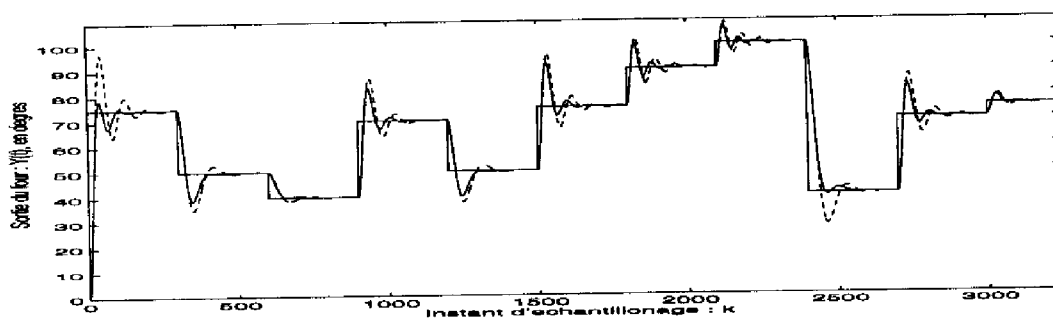


Figure 6.13 : Gain variable: régulateur flou standard (- -), régulateur flou adapté (-).

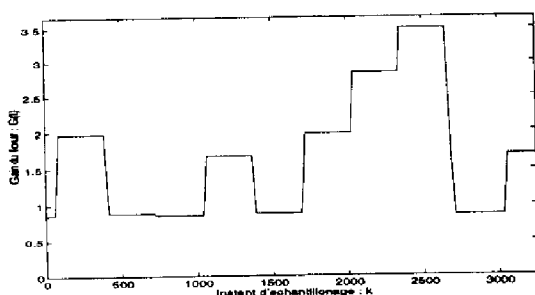


Figure 6.14 : Variations du gain K_s .

Il est possible d'observer un meilleur suivi de rampes, une amélioration notable des comportements oscillatoires et une rapidité accrue en début de régime transitoire avec le régulateur flou adapté. Ce dernier conduit la sortie du four avec moins d'oscillations sur les forts niveaux de consigne, malgré une augmentation importante de la valeur du gain.

Les indices de sensibilité sont donnés dans les tableaux (6.4) et (6.5). I_D représente, par exemple, l'indice de sensibilité de chacun des régulateurs pour la performance D .

Les signes entre parenthèses expriment des dégradations (-) ou des améliorations (+) de la performance considérée, entre une régulation des modèles nominal et perturbé.

Avec les rampes, le régulateur flou adapté est moins sensible aux perturbations sur le gain, pour ce qui est des oscillations: voir les dégradations du critère *temps de stabilisation* (ce critère est inférieur pour le régulateur PI, mais il faut tenir compte du plus grand amortissement de la réponse avec le régulateur flou adapté qui allonge cette mesure sans que ce soit lié à un comportement oscillatoire). Dans le cas des échelons, une plus grande sensibilité du régulateur flou adapté est mise en évidence avec le critère pourcentage de dépassement maximal; ce critère s'est dégradé trois fois plus qu'avec les autres régulateurs.

	PI	RFB ₀	RFB ₁
D (%)	20.5	21.5	11
t _{stab} (mn)	29	48	21.5
I _D (%)	(-)111.5	(-)69.5	(-)120
I _{t_{stab}} (%)	(-)18.5	(-)47.5	(-)26.5

Tableau 6.4 : Performances et sensibilité (rampes).

	PI	RFB ₀	RFB ₁
D (%)	24	21.5	15.5
t _m (mn)	4	3.9	4.1
I _D (%)	(-)243	(-)231	(-)812
I _{t_m} (%)	(+)55	(+)56	(+)58

Tableau 6.5 : Performances et sensibilité (échelons).

b- Variations de la constante de temps du four: La deuxième perturbation appliquée est la variation de la constante de temps du four selon la procédure décrite dans la section (1.4.2) du chapitre (1). Les résultats sont présentés sur les figures: (6.15), (6.16), (6.19) (rampes) et (6.17), (6.18), (6.19) (échelons).

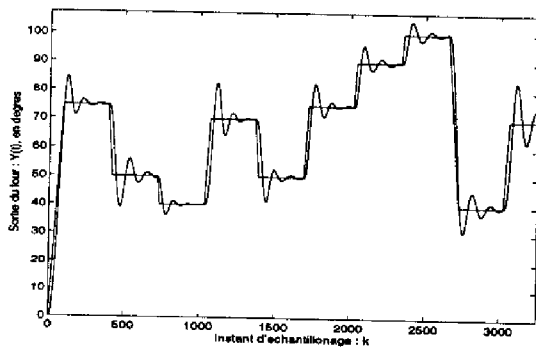


Figure 6.15 : T_s variable: PI.

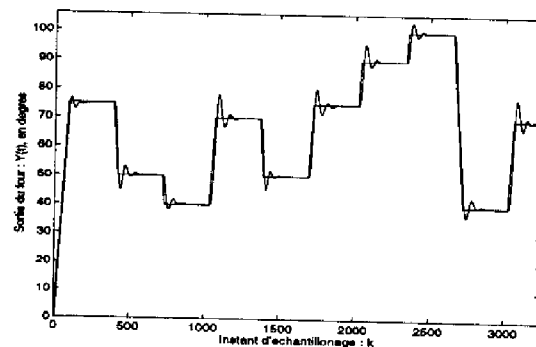


Figure 6.16 : T_s variable: RF adapté.

Le régulateur flou adapté conduit toujours aux meilleures performances: l'erreur de suivi de rampes est diminuée, les comportements oscillatoires sont notablement diminués et rendus acceptables et la rapidité est accrue en début de régime transitoire, par rapport aux autres régulateurs.

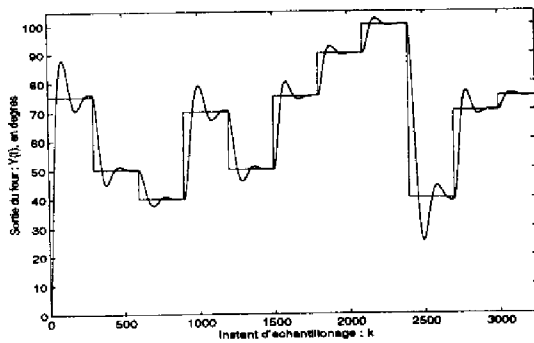


Figure 6.17 : T_s variable: PI.

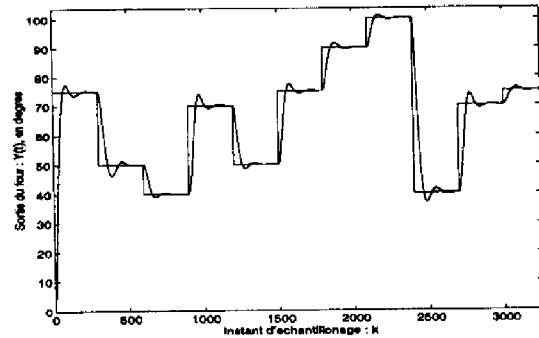


Figure 6.18 : T_s variable: RF adapté.

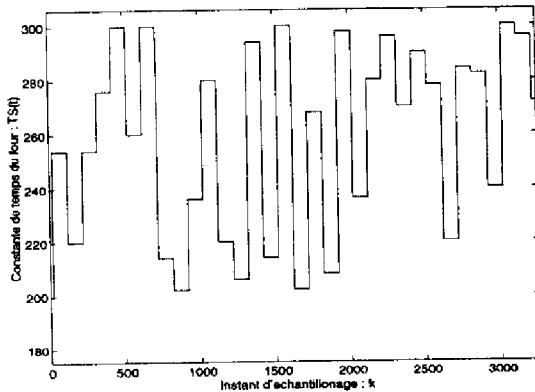


Figure 6.19 : Variations de T_s .

Les indices de sensibilité sont donnés dans les tableaux (6.6) et (6.7). La réponse du four est peu dégradée, par rapport à celle du four nominal, avec le régulateur flou adapté. Ce dernier est globalement moins sensible aux perturbations sur la constante de temps pour les deux types de consignes, malgré une dégradation supérieure du pourcentage de dépassement maximal, avec les échelons, que pour les deux autres régulations.

	PI	RFB_0	RFB_1
D (%)	14	16	7
t_{stab} (mn)	48	48	21
I_D (%)	(-)44.5	(-)26	(-)40
$I_{t_{stab}}$ (%)	(-)96	(-)46.5	(-)23.5

Tableau 6.6 : Rampes.

	PI	RFB_0	RFB_1
D (%)	9	9	4
t_m (mn)	10	10	10
I_D (%)	(-)28	(-)38	(-)135
I_{t_m} (%)	(-)11	(-)12	(-)2

Tableau 6.7 : Echelons.

c- Retard pur important non modélisé: La troisième cause d'introduction de non-linéarités dans la loi de régulation est un retard pur égal à 40% de la constante de temps, simulant une erreur de modélisation. Les résultats sont fournis sur les figures (6.20) (rampes) et (6.21) (échelons). Les courbes montrent une incapacité du régulateur flou adapté à affronter la situation dans le cas des rampes; le comportement observé est beaucoup trop oscillatoire. Le régulateur flou adapté ne conduit alors plus aux meilleures performances. Dans le cas des échelons, il permet d'obtenir les meilleures performances, tout en restant trop oscillatoire. Il apparait clairement que

le régulateur flou adapté, tel qu'il a été conçu, ne permet pas d'appréhender correctement des retards purs importants non modélisés.

L'utilisation de la formule (6.1) d'adaptation de ΔU_{max} en début de chaque régime transitoire permet d'améliorer notablement les résultats, voir les figures (6.20) et (6.21). Les indices de sensibilité sont donnés dans les tableaux (6.8) et (6.9). La réponse est maintenant acceptable, les oscillations ont disparu et la réponse est beaucoup plus amortie. Les pourcentages de dépassement maximaux sont bien inférieurs à ceux des autres régulateurs et très peu dégradés par rapport au cas nominal. En contrepartie, les temps de montée vers les échelons, ainsi que les erreurs de suivi de rampes sont augmentés de façon importante. Les pourcentages de dégradation des performances sont intéressants. La réponse du four, en présence d'un retard pur important non modélisé, est donc globalement améliorée par rapport aux cas d'utilisation des autres régulateurs, moyennant l'utilisation d'une formule simple d'adaptation de la puissance (nécessitant une estimation de la valeur du retard pur et du gain courant).

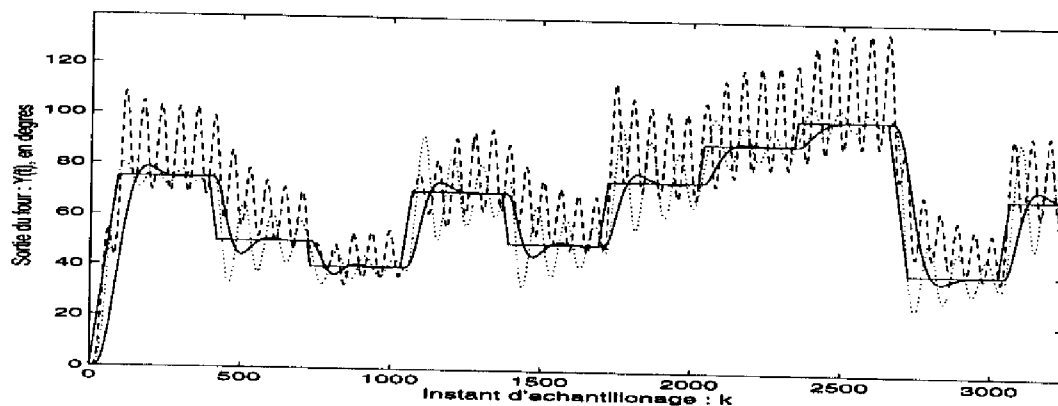


Figure 6.20 : PI (..), RF adapté (- -), RF avec base et puissance adaptée (-) (rampes).

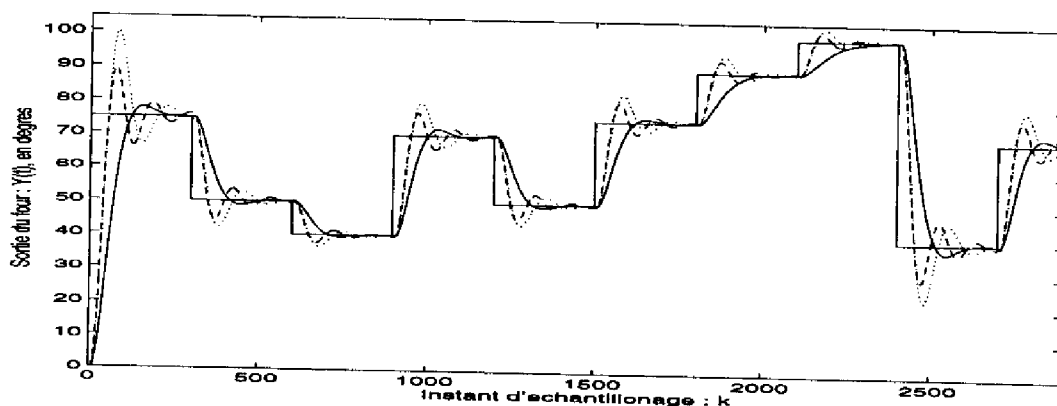


Figure 6.21 : PI (..), RF adapté (- -), RF avec base et puissance adaptée (-) (échelons).

	PI	RFB_0	RFB_1	$RFB1_{\Delta U}$
D_{max}	24	41	27	5.5
T_{stab}	50	50	50	38.5
$I_{D_{max}}$	-147.5	-223	-440	-10
$I_{T_{stab}}$	-104	-54	-194	-126.5

Tableau 6.8 : Rampes.

	PI	RFB_0	RFB_1	$RFB1_{\Delta U}$
D_{max}	16	16	11	2.3
T_m	9	8.4	7.8	22
$I_{D_{max}}$	-128	-146	-547	-35
I_{T_m}	0	+6	+20	-124

Tableau 6.9 : Echelons.

6.4.2 Simulations avec un auto-réglage autonome

Nous appliquons maintenant la procédure d'auto-réglage du paragraphe (6.3), qui permet de converger vers un bon compromis pour D et T_m , dans un domaine spécifié par l'utilisateur. Les dépassements et temps de montée à ne pas dépasser sont fixés respectivement à: $D_{min} = 3\%$ et $T_{min} = 900s$, le pas d'adaptation maximal des singletons est choisi égal à $PAS_{\Delta U_{max}} = 35\%$. La dynamique initiale RFB_0 est moins rapide que dans le cadre de l'auto-réglage précédent.

Nous donnons des courbes liées à la phase d'auto-réglage, puis à la phase de test. Pour caractériser le déroulement de l'auto-réglage, sont fournis les tracés des pas d'adaptation des singletons des conclusions des règles (inférés par le module d'adaptation), figure (6.22), des pourcentages de dépassements D , figure (6.23), et des temps de montée à 100% T_m , figure (6.24). Pour caractériser la qualité de l'auto-réglage, sont proposées des courbes obtenues pendant la phase de test, à savoir: les sorties du four avec les RF non adapté et adapté, figure (6.25), les évolutions du dépassement, figure (6.26), et les temps de montée, figure (6.27).

Les pas d'adaptation sont positifs pour diminuer le premier dépassement, puis négatifs pour corriger un comportement trop amorti, voir la figure (6.22). On observe une stabilisation de l'algorithme d'adaptation au bout de trois transitoires (15000s), puis un maintien des performances dynamiques (D, T_m) dans les zones de tolérance, voir les courbes (6.23) et (6.24). Concernant la phase de test, l'utilisation du RF adapté met en évidence des améliorations notables des performances, par rapport au RF de base, par réduction du comportement oscillatoire, voir la courbe (6.25). Le tableau (6.10) contient les performances dynamiques moyennes atteintes. Les dépassements diminuent au détriment, légèrement, des temps de montée.

Il est possible de vérifier expérimentalement, que les performances ne sortent pas du domaine de spécifications défini par l'utilisateur, voir les courbes (6.26) et (6.27).

	RFB_0	RFB_1
$D(\%)$	5	1.3
$T_m(mn)$	8.8	9.8

Tableau 6.10 : Performances.

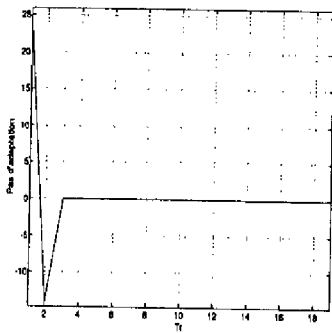


Figure 6.22 : Adaptation des singletons.

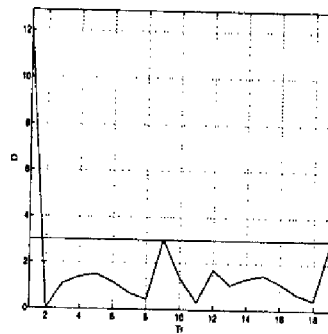


Figure 6.23 : $D(\%)$ pendant l'adaptation.

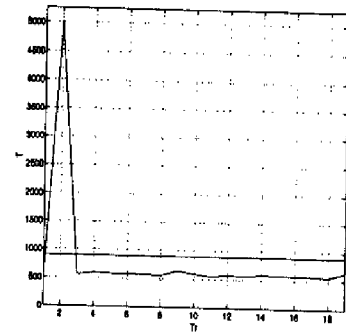


Figure 6.24 : T_m (s) pendant l'adaptation.

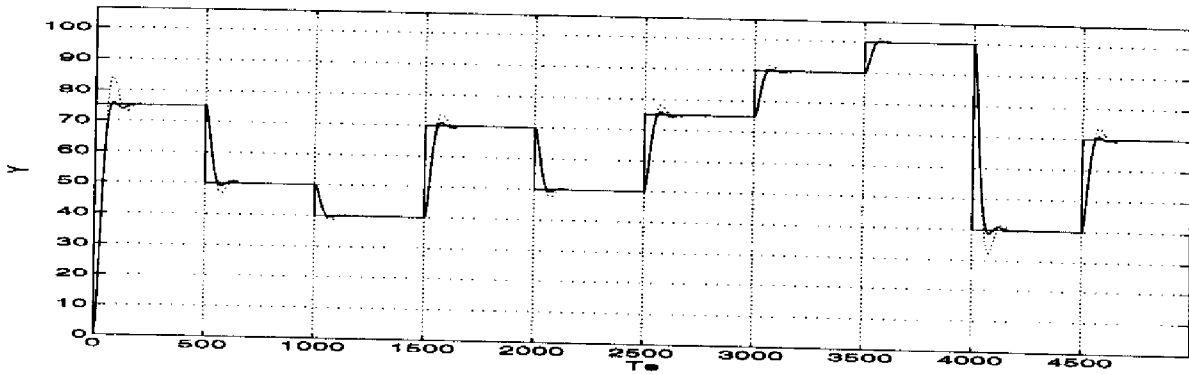


Figure 6.25 : Régulation après l'auto-réglage: RFB_0 (..), RF adapté (.)

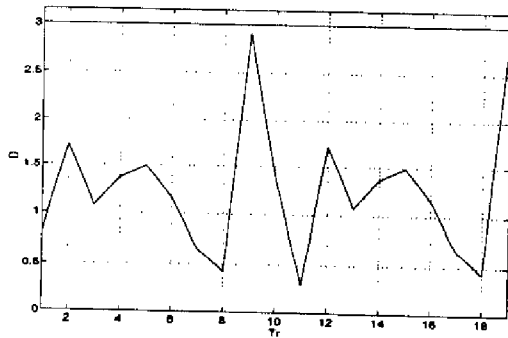


Figure 6.26 : D (%) après l'adaptation.

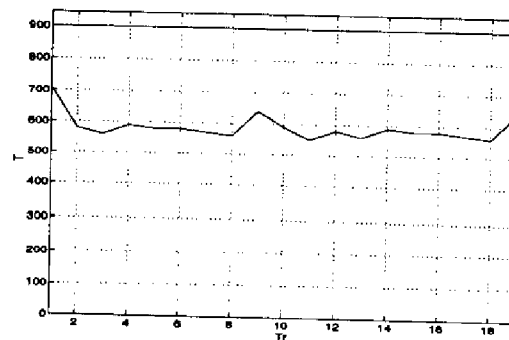


Figure 6.27 : T_m (s) après l'adaptation.

6.4.3 Auto-réglage de base avec la réglette à infra rouge

Les tests menés sur des rampes et des échelons visent à comparer les résultats d'une régulation floue, à ceux d'une régulation PID industrielle. Les PID ont été réglés par des spécialistes industriels (rampes: $X_p = 20$, $T_i = 0$ mn, $T_d = 27$ mn, $CAS = 75$, échelons: $X_p = 40$, $T_i = 10$ mn, $T_d = 8$ mn, $CAS = 50$). Les RF ont pour paramètres:

pour les rampes:

- $K_p = 1.75, T_i = 200$ s, avec la base de règles standard ($RFB0^r$),
- $K_p = 1.75, T_i = 200$ s, avec la base adaptée (décalage de 0.25) ($RFB1^r$),
- $K_p = 1, T_i = 300$ s, avec la base adaptée (0.225) ($RFB2^r$);

pour les échelons:

- $K_p = 2.5, T_i = 50$ s, avec la base de règle standard ($RFB0^e$),
- $K_p = 2.5, T_i = 50$ s, avec la base adaptée (0.1) ($RFB1^e$).

A- Poursuites de rampe

Les figures (6.28) à (6.30) présentent les résultats. L'observation des deux premières figures permet de souligner l'amélioration des performances apportées par l'adaptation de la base de règles ($RFB0^r$ et $RFB1^r$): pour un temps de stabilisation équivalent et un meilleur suivi de rampe, le dépassement est réduit de 2.2°C à 1.6°C .

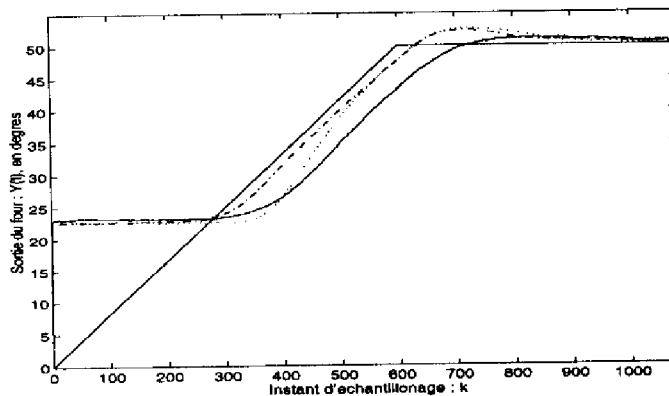


Figure 6.28 : $RFB0^r$.., $RFB1^r$ --, $RFB2^r$..

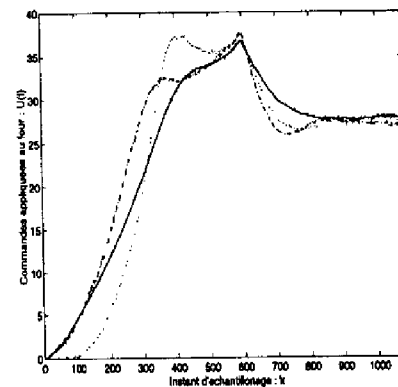


Figure 6.29 : Commandes.

La dernière figure montre les améliorations apportées par rapport à la régulation PID industrielle.

Dans le premier cas ($RFB1^r$), le suivi de rampe est notablement amélioré (amélioration de +83% en début de rampe et de +50% en fin de rampe), pour des dégradations du dépassement de (-31%) (1.6°C à 1.1°C) et du temps de stabilisation de (-34%). Dans le second cas ($RFB2^r$), le dépassement est amélioré (+27%) (1.1°C à

0.8°C) et le suivi de rampe est équivalent en début et détérioré en fin (-25%), pour une dégradation du temps de stabilisation d'environ -34%. Dans les deux cas, la dégradation du temps de stabilisation n'est pas due à un comportement trop oscillatoire, mais à un plus grand amortissement de la sortie du four, à cause de l'anticipation plus importante sur les dépassements.

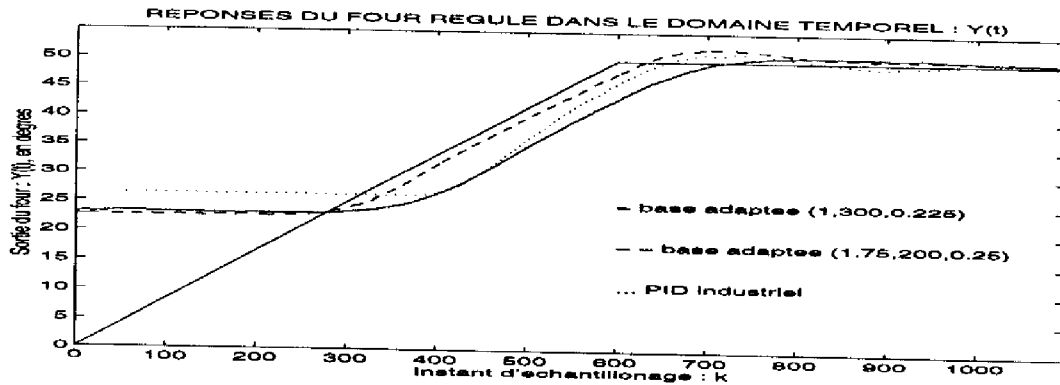


Figure 6.30 : Comparaison au PID industriel ..., $RFB1^r$ --, $RFB2^r$..

B- Réponse à un échelon

Les figures (6.31) à (6.33) présentent les résultats. Les deux premières figures visualisent l'amélioration des performances apportées par l'adaptation de la base de règles: le dépassement est annulé pour une détérioration du temps de montée de -71% (140s à 240s), causée par un amortissement important en fin de régime transitoire.

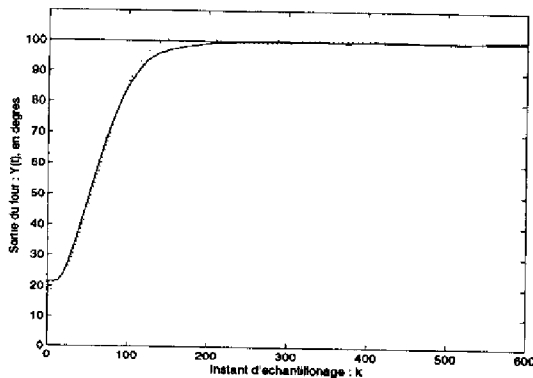


Figure 6.31 : $RFB0^e$..., $RFB1^e$..

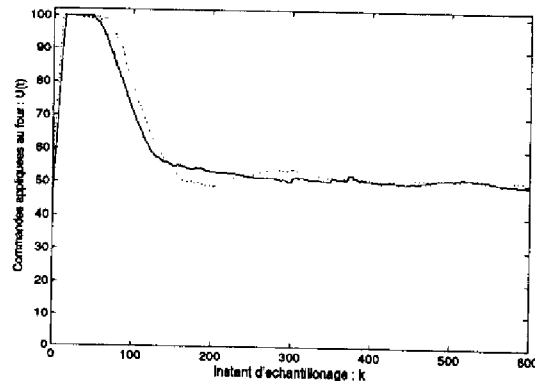


Figure 6.32 : Commandes.

Les améliorations suivantes sont observées par rapport à la régulation PID industrielle, voir la courbe (6.33): le comportement dynamique en boucle fermée est plus rapide (amélioration du temps de montée de +17%), pour une condition initiale plus contraignante et un dépassement nul dans les deux cas.

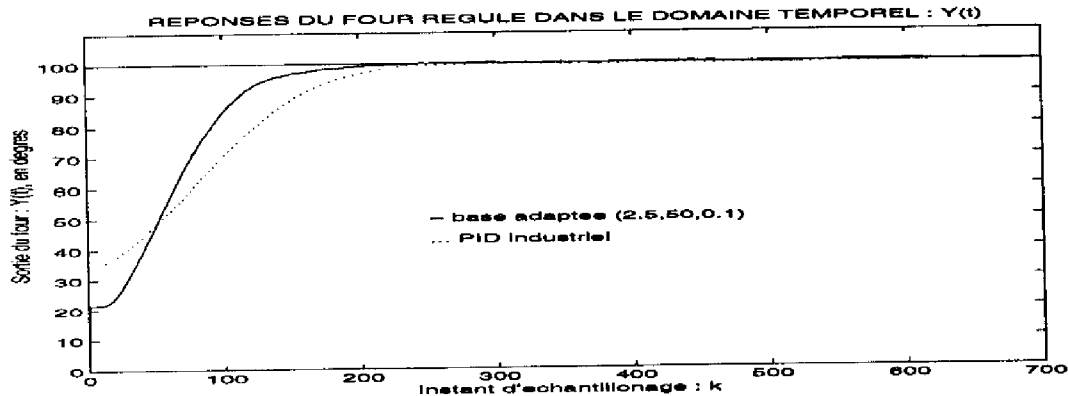


Figure 6.33 : Comparaison au PID industriel .., $RFB1^e$..

Les résultats des expérimentations sur la réglette à infra rouge sont résumés dans les tableaux (6.11) et (6.12), exprimant les différences par rapport aux performances dynamiques obtenues avec la régulation PID industrielle. Nous pouvons conclure que des améliorations intéressantes ont été apportées par les régulateurs flous proposés.

	Diff.D	Diff.T _{stab}	Diff.suivi
$RFB1^e$	-31%	-34%	PG
$RFB2^r$	+27%	-34%	NP

Tableau 6.11 : Différences avec la régulation PID (rampes).

	Diff.D	Diff.T _m
$RFB1^e$	0%	+17%

Tableau 6.12 : Différences avec la régulation PID (échelons).

6.5 Conclusions et extensions

A- Conclusions

Nous avons développé une procédure de conception de RF en deux étapes. La structure de base du bloc de commande est celle d'un régulateur flou de type PI linéaire, conduisant à une dynamique rapide en boucle fermée. Des non-linéarités sont alors introduites dans le régulateur flou, de façon à optimiser les performances dynamiques. Ces non-linéarités sont de deux types.

D'une part, elles sont intégrées dans la base de règles de façon à réduire les dépassements, tout en assurant des performances quasi-équivalentes en terme de rapidité. Deux procédures de réglage ont été proposées pour adapter automatiquement la base de règles. Ces procédures sont lancées sur un cycle de référence avant la régulation définitive, soit sur le four à réguler, soit sur un modèle de ce four.

Par ailleurs, d'autres non-linéarités sont introduites sous forme de modulation de la puissance maximale disponible à chaque début de régime transitoire, de façon à obtenir un bon comportement en présence d'un retard important non-modélisé.

L'auto-réglage du régulateur flou n'est pas difficile à mettre en oeuvre; il ne nécessite pas le choix heuristique de beaucoup de paramètres. Son utilisation est facile pour un industriel familier du PID. Les deux facteurs de la dynamique PI initiale peuvent être déduits directement d'une expérience automatique de pré-réglage par la méthode du relais. Les trois paramètres d'adaptation de la base de règles sont fixés d'après les performances désirées, ils sont liés à un modèle de référence du comportement en boucle fermée. La convergence n'est toutefois pas garantie pour n'importe quels paramètres. Des choix judicieux sont nécessaires à partir d'une connaissance initiale sur la dynamique du four et par modifications successives de valeurs des spécifications non-réalistes.

La démarche proposée permet de générer, selon une approche simple, une loi de régulation non linéaire qui, appliquée à un système linéaire, permet d'obtenir un compromis *rapidité/dépassements* très intéressant. Dans le cas des simulations avec un modèle du second ordre, les tests de sensibilité avec les régulateurs flous adaptés sont satisfaisants, particulièrement dans le cas de perturbations sur la constante de temps du système. Toutefois, dans le cas du four possédant un retard pur important non modélisé, il est nécessaire d'introduire une adaptation supplémentaire. Des résultats intéressants ont été obtenus également en temps réel et des améliorations ont été notées par rapport au régulateur PID industriel.

B- Extensions

Les bonnes propriétés dynamiques obtenues sont un encouragement pour envisager l'introduction d'autres non-linéarités dans la loi de régulation floue. Dans ce but, il est possible d'adapter les règles liées au dépassement de façon différenciée, d'adapter d'autres règles et d'introduire des stratégies d'adaptation particulières suivant qu'il s'agit d'un problème de poursuite ou de régulation.

Il est aussi envisageable de construire un module d'adaptation entièrement autonome. Il pourrait démarrer avec un $PAS_{\Delta U_{max}}$ très petit, qui serait augmenté progressivement jusqu'à ce que les performances atteintes, sur un horizon donné de simulation, entrent dans les domaines spécifiés par l'utilisateur ou jusqu'à une limite de stabilité du four. La procédure pourrait ainsi fournir une valeur acceptable du pas d'adaptation maximal ou une indication de l'impossibilité de rejoindre les spécifications fournies.

Une des étapes suivantes de notre travail pourrait consister à mener à terme des études expérimentales comparatives de robustesse en stabilité et en performances plus poussée, en cherchant les limites des fonctionnements désirés, pour diverses variations des paramètres internes du four, ceci afin de préciser, plus finement, les qualités fondamentales du régulateur non linéaire généré.

D'autre part, des études de stabilité de la boucle fermée, complétée par le module d'adaptation, s'avèrent nécessaires.

Conclusion

L'objectif général de notre travail était l'évaluation de techniques floues et neuronales pour des problèmes de régulation bas niveau ou adaptative, afin d'en déduire des possibilités de développement d'outils efficaces pour l'Automatique. Pour mener à bien notre travail, plusieurs applications ont été envisagées: un pendule inversé propre au laboratoire et des modèles de fours utilisés dans le cadre d'une coopération industrielle. Nos études ont donc été orientées vers des caractérisations expérimentales, ainsi que vers des aspects méthodologie de conception.

La synthèse des travaux réalisés permet de présenter diverses réalisations qui sont: un régulateur flou avec inférence hors ligne réglé de façon heuristique (pour le pendule inversé), un RF avec inférence en ligne réglé d'après les lois d'équivalence avec un régulateur PI linéaire (pour les fours), une structure de régulation neuronale directe utilisant l'apprentissage spécialisé par rétropropagation (pour le pendule inversé), deux superviseurs flous de régulateurs PID académique ou industriel (pour les fours) et un RF réglé avec un système flou d'auto-réglage (pour les fours industriels).

A- Conclusions

a- Apports des techniques floues

L'utilisation de systèmes experts flous permet une nouvelle approche de conception de régulateurs (utilisation de connaissance linguistique permettant d'intégrer la connaissance experte). Derrière le raisonnement approché, apparaît un interpolateur non linéaire (approximateur universel). L'introduction nécessaire de non-linéarités dans la loi de régulation pose le problème du réglage des nombreux paramètres des systèmes flous.

Nos apports, au niveau méthodologique, sont indiqués ci-dessous:

1. Nous avons, tout d'abord, mis en oeuvre un RF avec table de décision. Pour diminuer les erreurs d'approximation numérique de la loi de régulation liées au mécanisme de quantification, nous avons proposé une interpolation pour la mise en oeuvre en ligne. Cette méthode est plus simple et plus efficace que l'utilisation de plusieurs niveaux de réglages fin et grossier.

2. Nous avons aussi développé un RF avec inférence en ligne, en appliquant une méthode d'auto-réglage, pour introduire facilement des non-linéarités dans la loi de commande et optimiser la forme des réponses temporelles. L'utilisation des règles d'équivalence pour générer une loi de régulation PI quasi-linéaire conduit, tout d'abord, à un *régulateur flou standard* (à ce niveau sont choisis une dynamique rapide à travers les gains proportionnel et intégral et le niveau de saturation). La modification automatique de certaines règles de régulation floue permet d'atténuer les comportements oscillatoires et conduit à un *régulateur flou adapté* avec des non-linéarités intervenant dans certaines zones du plan de phase. Nous avons rendu la technique d'adaptation floue des conclusions des règles plus autonome, en proposant la définition d'une zone de non adaptation des règles (en dessous de performances limites tolérées) et une optimisation bidirectionnelle sur l'univers de discours de ΔU .

Eventuellement, une dernière étape peut consister à modifier la loi de régulation en utilisant les interpolations liées au mécanisme d'inférence. Les effets non linéaires correspondants sont plus ou moins locaux et, donc, plus ou moins sensibles en fonction du nombre de règles utilisées.

3. De plus, nous avons mis en oeuvre deux superviseurs flous de PID selon une approche simple: une seule table de règles avec trois formules d'adaptation, permet de déduire les modifications à appliquer aux gains du régulateur, en fonction de la position dans le plan de phase. Une des structures proposées inclut des garanties de stabilité. Nous avons proposé des liens entre paramètres de conception et performances en boucle fermée.

Il est possible de rapprocher les deux dernières techniques développées pour générer des lois de régulation non linéaires. Dans les deux cas, des modifications locales sont introduites sur la surface de commande de type PI.

Pour le RF, les changements sont appliqués par action directe sur les niveaux de commande, dans des zones qualitatives spécifiques de l'espace de phase. Les changements sont automatiques à partir des performances dynamiques. Dans le cas du SF, des actions indirectes sont réalisées à travers les variations des gains proportionnel et intégral, en utilisant le savoir de l'ingénieur automaticien.

La deuxième approche apparaît plus attirante pour un ingénieur maîtrisant le PID. Toutefois la première approche est plus homogène et immédiate (en effet, dans le deuxième cas, il est nécessaire de reconstruire la loi de régulation).

Avec les différentes structures de commande mettant en oeuvre un système expert flou, de bonnes performances ont été obtenues en boucle fermée. Les techniques floues sont donc des aides efficaces pour améliorer, en ligne, les résultats de régulations conventionnelles, en effet:

1. Des comparaisons, avec des résultats de la littérature sur le pendule inversé, nous ont amenée à conclure que le RF standard conduisait à de meilleures performances (surtout en termes de temps de montée) que des régulateurs linéaires

conventionnels. Les non-linéarités *naturelles* du RF (saturations et interpolations non linéaires liées au processus d'inférence) expliquent ces résultats.

2. Certains effets bénéfiques dus aux interpolations non linéaires entre les points caractéristiques (points modaux et points de recouvrement à 0.5), ont parfois été mis en évidence, avec des consignes en échelon ou en rampe pour les fours (principalement en régime transitoire et parfois aussi avec les dépassements en régime permanent).
3. Des améliorations importantes ont été apportées avec le RF auto-réglé (utilisé sur les fours) par rapport au régulateur PI linéaire et au RF standard. De plus, des indices de sensibilité intéressants ont été obtenus. Nous avons également noté des performances améliorées par rapport au régulateur PID industriel (en terme d'atténuation des dépassements et, dans une moindre mesure, au niveau des régimes transitoires).
4. Concernant les SF de PID académique ou industriel, des améliorations notables ont été observées sur les fours par rapport au régulateur non adapté (surtout pour le régime transitoire et la rapidité du système de commande).

b- Apports des techniques neuronales

Les réseaux de neurones permettent de générer, de façon adaptative, de la connaissance à partir de données expérimentales. Ils possèdent la propriété intéressante d'être des approximateurs universels et, donc, de pouvoir conduire à des lois de commande non linéaires.

Nous nous sommes intéressée à la structure de régulation neuronale directe avec apprentissage spécialisé. Nous avons développé une stratégie de définition automatique de lois de régulation de type *régime glissant*, à partir d'une connaissance initiale quasi-nulle sur le système et sur la connaissance à intégrer dans le régulateur. Nous avons montré que le choix du critère d'apprentissage permet de spécifier les performances dynamiques attendues. Le réglage automatique du RN permet donc de converger vers un modèle de référence. Des méthodes pour améliorer l'apprentissage ont été proposées, elles concernent une meilleure approximation du jacobien et la prise en compte de la dynamique du système.

Nous avons vérifié en simulation, avec le pendule inversé, que la stratégie de régulation pré-définie était effectivement apprise. Sur le système réel, nous avons pu stabiliser la tige et le chariot du pendule; toutefois des difficultés d'apprentissage sont apparues. La stabilisation de la seule tige peut, par contre, être réalisée de façon très rapide.

B- Prospective

De nombreuses recherches peuvent être continuées en suivant les voies parcourues dans ce travail. Les RF, RN, SF ont des potentialités intéressantes de robustesse dues à leur non-linéarités. Les résultats prometteurs obtenus laissent entrevoir des possibilités d'amélioration supplémentaires, par introduction de non-linéarités additionnelles dans la loi de commande.

Le tableau récapitulatif (6.13) permet de mesurer certains points forts et certaines limitations des approches préconisées. Le tableau (6.13) est une base pour définir les extensions possibles de notre travail...

	RF	RN	SF
Résultats expérimentaux	++	///////	+
Si beaucoup de connaissance expérimentale disponible	+	++	-
Interface linguistique pour le concepteur et l'utilisateur	++	-	+
Pour des industriels familiers avec le PID	+	-	++
Preuves de stabilité	-	-	++
Pré-définition d'un modèle de référence	+	++	-
Facilité de conception	++	-	+
Type de systèmes	<i>système non linéaire d'ordre limité</i>	<i>système non linéaire de bas ordre</i>	<i>système non linéaire d'ordre limité (incertitudes structurées)</i>

Tableau 6.13 : Tableau comparatif des méthodes employées.

Les trois structures de commande sont comparées, selon différents critères. Nous ne comparons les résultats expérimentaux que dans les cas de contextes équivalents (cas des modèles de fours). Les limitations sur les types de systèmes sont liées à:

pour les RF: l'utilisation d'une structure **PI**, éventuellement insuffisante pour des ordres élevés;

pour les RN: la définition d'une droite de glissement adaptée pour un ordre 2;

pour les SF: le type de non linéarité imposé est lié au critère de stabilité.

a- Aspects court terme

Concernant le RF auto-réglé, il pourrait être envisagé d'adapter de façon différenciée les règles liées au dépassement (et non plus systématiquement dans une même proportion), de façon à affiner les résultats. D'autre part, des règles liées à d'autres performances (situées dans d'autres régions du plan de phase) pourraient également être adaptées. Des stratégies particulières pourraient être établies selon le type de consigne (rampes ou échelons).

A propos du SF, il serait judicieux d'affiner la stratégie de supervision en modifiant localement la table de règles, éventuellement par un système automatique.

Dans le cas du RN, il faudrait, dans un premier temps, étudier plus précisément la stratégie de régulation apprise sur le système réel. Des modifications pourraient également être apportées à la démarche de conception: meilleur apprentissage en calculant plus précisément le gradient (en adaptant les poids du réseau en concordance avec la dynamique du système ou en utilisant d'autres formes d'estimation du jacobien du processus -ce qui est un point essentiel pour la régulation de θ et y -).

b- Prospective à plus long terme

Les principaux axes de recherche proposés sont, tout d'abord, des couplages de techniques floues et neuronales permettant, surtout, de compléter automatiquement la connaissance haut-niveau, à partir de données uniquement expérimentales.

De plus, la maîtrise des interpolations non linéaires dues à l'inférence entre les points caractéristiques pourrait permettre d'établir, de façon pertinente, des bases de règles minimales. Les effets dus à l'inférence sont, en effet, d'autant moins locaux, et d'autant plus importants, que le nombre d'ensembles flous est réduit. L'obtention de connaissance linguistique minimale est effectivement un des enjeux principaux pour le développement de régulateurs flous destinés à contrôler des systèmes complexes. Les règles graduelles sont peut être une voie intéressante. Outre une interprétation logique bien fondée et l'existence de théorèmes de vérification de la cohérence de la base de règles, elles permettent aussi de réaliser des interpolations non linéaires. Un point primordial reste les études de *stabilité* pour le RF de la seconde génération.

Ensuite, la combinaison des deux techniques de supervision floue permettrait de mener des adaptations, en fonction de la position dans le plan de phase et des performances dynamiques atteintes au cours du dernier régime transitoire. Cette deuxième approche permettrait de spécifier plus naturellement un modèle de référence. Ceci nous conduit à nous pencher sur les aspect *interface Homme-machine*; car il ne faut pas oublier que les systèmes experts flous sont particulièrement efficaces pour les tâches de raisonnement, donc pour les niveaux plus hauts des SNCC.

Finalement, après avoir établi des preuves de stabilité pour l'approche de régulation neuronale utilisée, d'autres stratégies de commande pourraient être développées, via la recherche d'autres critères ou par le choix d'architectures plus complexes.

Beaucoup de travail reste donc à faire pour valider complètement les techniques floues et neuronales utilisées en Automatique, car les façons d'appréhender les problèmes de commande sont nouvelles. Toutefois, certaines spécificités de ces approches en font, d'ores et déjà, des compléments indispensables des méthodes analytiques largement développées.

Références

- [AST,86] Astrom, K.J., Anton, J.J., Arzen, K.E., "Expert Control", *Automatica*, vol. 22 num. 3, 1986, p. 277-286.
- [AST,89] Astrom, K.J., Wittenmark, B., "Adaptive Control", Ed. Addison-Wesley, 1986.
- [AST,92] Astrom, K.J., "Autonomous Controllers", IFAC/IFIP/IMACS (*International Symposium on Artificial Intelligence in Real-Time Control*), Delft Pays Bas, 16-18 Juin 1992, p. 1-6.
- [BAB,92] Babuska, R., Horacek, P., "Laboratory Evaluation of fuzzy Controllers", IFAC/IFIP/IMACS, Delft Pays Bas, 16-18 Juin 1992, p. 263-268.
- [BER,94] Bersini, H., Gorrini, V., "Local Interpolation, global Interpolation, fuzzy Systems and neural Networks for non-linear Function Approximation", FALCON (*Fuzzy Algorithms for Control*), Aachen Allemagne, 19 Septembre 1994.
- [BOU,93] Bouchon-Meunier, B., "La logique floue", Que sais-je ?, Ed.: Presses universitaires de France, ISBN 2130450075, 1993.
- [BOY,95] Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V., "Linear Matrix Inequalities in System and Control Theory", 15 Siam Studies in Applied Mathematics, 1995.
- [CUL,95] Culliere, T., Titli, A., Courrieu, J.M., "Neuro-Fuzzy Modeling of non-linear Systems for control purpose", FUZ-IEEE, Japan, 20-24 Mars 1995, p. 2009-2016.
- [DEG,91] De Geest, D., "Predictive Control of a nonlinear Process using Neural Networks", mémoire de stage DEA-Ingénieur INSAT, INSAT (Toulouse France) et TUD (Delft Pays Bas), Septembre 1991.
- [DEG,93] De Geest, D., Rehwald, W., Futterlieb, E., Titli, A. "Comparison of Algorithmic, Fuzzy and Neural Controllers based on Applications to an inverted Pendulum", EUFIT (*European Congress on Fuzzy and Intelligent Technologies*), Aachen Allemagne, Septembre 1993, p. 443-449.
- [DEG,94a] De Geest, D., Rehwald, W., Futterlieb, E., Titli, A. "Applications of Fuzzy and Neural Controllers on a Laboratory Equipment", AE in ACT (*International TEMPUS Workshop on Advanced Education in Automation and Control Technology*), Prague République Tchèque, 16-18 Mai 1994, p. 74-77.
- [DEG,94b] De Geest, D., Titli, A., Flacher, M., "Régulation floue de Fours industriels", Journées nationales sur les Ensembles Flous, Lille France, Novembre 1994, p. 57-66.
- [DEG,94c] De Geest, D., Ketata, R., Jocteur Monrosier, H., Titli, A., Flacher, M., "Développement d'un régulateur flou, d'un superviseur flou de PID et d'une structure de commande parallèle flou/PID - Applications à des fours industriels", Rapport LAAS n. 94510, Décembre 1994.
- [DEG,95a] De Geest, D., Titli, A., "Régulation floue et neuronale d'un pendule inversé", Journées nationales sur: *l'Automatique et les expérimentations en laboratoire* (club EEA), Angers France, 4-5 Avril 1995, p. 4.1-4.10.
- [DEG,95b] De Geest, D., Titli, A., "Neural Control with specified closed-loop dynamics", Third EUFIT conference, Aachen Allemagne, 26-29 Aout 1995, p. 815-822.

- [DON,93] Don, R., Horne, G., "Progress in Supervised Neural Networks - What's new since Lippman?", IEEE signal processing magazine, Janvier 1993, p. 8-39.
- [DUB,90] Dubois, D., "Automatique et Intelligence artificielle: pour un Dialogue", La Lettre de l'Automatique, num. 8, Octobre 1990, p. 2-3.
- [DUB,91] Dubois, D., Prade, H., "Systèmes à base de règles floues et leurs applications pour la décision et la commande", Rapport IRT/91-92/R, Décembre 1991.
- [EFS,92] Efstathiou, J., "Architectures and Techniques of Artificial Intelligence in Process Control", IFAC/IFIP/IMACS, Delft Pays Bas, 16-18 Juin 1992, p. 479-484.
- [FOU,89] Foulloy, L., "Using qualitative Reasoning to write expert Controllers", Revue d'Intelligence Artificielle, Vol. 3 num. 4, 1989, p. 151-173.
- [FOU,92] Foulloy, L., Galichet, S., "Représentation des Contrôleurs flous" et "Contrôleurs flous: Représentation, Equivalences et Etudes comparatives", Journées nationales *Les Applications des Ensembles flous*, Nîmes France, 2-3/11 92, p. 129-136 et p. 229-236.
- [FOU,93] Foulloy, L., "Qualitative control and fuzzy control: towards a writing methodology", AI Communications, vol.6, no.3-4, p. 147-54.
- [FUT,93] Futterlieb, E., "Evaluation d'un régulateur neuronal: Application sur un Pendule Inversé", Rapport de stage *Erasmus*, LAAS Toulouse France, 1993.
- [GAR,94] Garcia-Cerezo, A., Ollero, A., "Design of stable fuzzy control systems from experimental data", EUFIT Conference, Aachen Allemagne, 20-23 Septembre 94, p. 1175-1182.
- [GEN,94] Gentil, S., "Knowledge Representation: Special Issues", Ecole d'été d'Automatique *Intelligence Artificielle et Automatique*, tome 1, Grenoble France, Septembre 1994.
- [GUP,86] Gupta, M.M., Trojan, G.M., Kiszka, J.B., "Controllability of Fuzzy Control Systems", IEEE Trnas. on Systems, Man and Cybernetics, vol. SMC-16 No 4, Juillet/Aout 1986, p. 576-582.
- [HAY,91] Hayashi, S., "Auto-tuning fuzzy PI Controller", IFSA91-Eng (*International Fuzzy Systems Association*), 1991, p. 41-44.
- [HUN,92] Hunt, K.J., Sbarbaro, D., Zbikowski, R., Gawthrop, P.J., "Neural Networks for Control Systems - A Survey", *Automatica*, vol. 28 No 6, 1992, p. 1083-1112.
- [JOC,93] Jocteur Monrosier, H., "Supervision par Logique floue d'un régulateur PID de Four industriel", Rapport de stage Ingénieur INSAT, Toulouse France, 1993.
- [JOR,89] Jordan, M.I., "Generic Constraints on underspecified Target Trajectories", Int. Joint Conf. on *Neural Networks*, vol. 1, Washington Etats Unis, 1989, p. 217-225.
- [KAU,75] Kaufmann, A., "Introduction to Theory of fuzzy Subsets", New York Ac., 75.
- [KET,92] Ketata, R., "Méthodologies de Régulation numérique incluant la Logique floue", Thèse de Doctorat I.N.S.A.T., Toulouse, Juillet 1992.
- [KET,93] Ketata, R., De Geest, D., Titli, A., "Fuzzy Controller: Design, Evaluation, Parallel and Hierarchical Combination with a PID Controller", fifth IFSA Conference, Seoul Corée, Juillet 1993, p. 814-817.
- [KET,95] Ketata, R., De Geest, D., Titli, A., "Fuzzy Controller: Design, Evaluation, Parallel and Hierarchical Combination with a PID Controller", *Fuzzy Sets and Systems*, special issue on *Fuzzy Control*, vol. 71, 1995, p. 113-129.
- [LEE,90] C.C. Lee, "Fuzzy logic in Control Systems : fuzzy logic Controller - part 1 and 2", IEEE Trans. Syst. Man. Cybern., vol. 20 n°. 2, Mars/Avril 1990, p. 404-435.
- [LEI,92] Leitch, R., "Knowledge Based Control: selecting the right Tool for the Job", IFAC/IFIP/IMACS, Delft Pays Bas, 16-18 Juin 1992, p. 25-33.

- [LIP,87] Lippman, R.P., " An introduction to computing with Neural Nets ", IEEE ASSP magazine, Avril 1987, p. 4-22.
- [LIT,91] Litt, J. , " An expert System to perform on-line Controller Tuning ", IEEE Control Systems, Avril 1991, p. 18-23.
- [MAC,76] MacVicar-Whelan, P.J., " Fuzzy Sets for Man-Machine Interaction ", Int. J. Man-Machine Studies, vol. 8, 1976, p. 687-697.
- [MAM,75] Mamdani, E.H., Assilian, S., " An experiment in linguistic Synthesis with a fuzzy logic Controller ", Int. J. Man-Machine Studies, vol. 7, 1975, p. 1-13.
- [MAR,83] Margulies, F., Zemanek, H., " Man's Role in Man-Machine Systems ", Automatica, vol. 19 n°. 6, 1983, p. 677-683.
- [MAR,94] Marin, J.P., Titli, A., " Comparative Analysis of Stability methods for fuzzy Controllers ", EUFIT Conference, Aachen Allemagne, Septembre 94, p. 1183-1187.
- [MAR,95] Marin, J.P., Titli, A., " On the use of describing Function to analyse and design fuzzy Controllers ", IFSA95, Sao-Paulo Bresil, Juillet 1995, p. .
- [MIZ,88] Mizumoto, M., " Fuzzy Controls under various fuzzy reasoning Methods ", Information Sciences, vol. 45, 1988, p.129-151.
- [MOR,76] Mori, S., Nishihara, H., Furuta, K., " Control of unstable mechanical System: Control of Pendulum ", Int. J. Control, Vol. 23 n°. 5, 1976, p. 673-692.
- [NAR,90] Narendra, K.S., Parthasarathy, K., " Identification and Control of dynamical Systems using neural Networks ", IEEE Trans. on *Neur. Net.*, vol. 1 n°. 1, Mars 90, p. 4-27.
- [NAU,90] van Nauta Lemke, H.R. and Krijgsman, A.J., " Design of fuzzy P.I.D. Supervisor for Systems with different Performance Requirements ", IMACS-91 (*International conference on Computation and Applied Mathematics*), Juillet 22-26 1991, Dublin Irlande, p. 1200-1203.
- [OLI,91] Oliveira, P., Lima, P., Sentieiro, J., " Fuzzy Supervision on intelligent Control Systems ", ECC, Grenoble France, 2-5 Juillet 1991, p. 1226-1231.
- [PSA,87] Psaltis, D., Sideris, A., Yamamura, A., " Neural Controllers ", IEEE Int. Conf. on *Neural Networks*, vol. 4, San Diego Etats Unis, 1987, p. 551-558.
- [REN,93] Renders, J.M., Bersini, H., Saerens, M., " Adaptive Neuro-Control: How black Box and simple can it be ", rapport technique n°. 93-7, IRIDIA, Université libre de Bruxelles Belgique, 1993.
- [REH,93] Rehwald, W., " Evaluation d'un régulateur flou: Application sur un Pendule Inversé ", rapport de stage *Erasmus*, LAAS Toulouse France, 1993.
- [ROD,92] Rodd, M.G., Verbruggen, H.B., Krijgsman, A.J., " Artificial Intelligence in Real-time Control ", Eng. Applic. Artif. Intel., vol. 5 n°. 5, 1992, p. 385-399.
- [RUG,94] Ruger, J.J., Frenck, Ch., Michalske, A., Kiendl, H., " The Vector Field Method for Stability Analysis: a case study ", EUFIT Conference, Aachen Allemagne, 20-23 Septembre 94, p. 291-302.
- [RUM,88] Rumelhart, D., Mc Clelland, J.L., " Exploration in Parallel Distributed Processing ", MIT Press, 1988.
- [SAE,91] Saerens, M., Soquet, A., " Neural Controller based on back-propagation Algorithm ", IEEE Proceedings-F, Vol. 138 n°. 1, Fevrier 1991, p. 55-62.
- [SAE,93] Saerens, M., Renders, J.M., Bersini, H., " Neural Controller based on back-propagation Algorithm ", rapport No 93-16, IRIDIA, ULB Belgique, 1993.
- [SCH,85] Scharf, E.M., Mandic, N.J., " The Application of a fuzzy Controller to the Control of a multi-degree-freedom Robot Arm ", dans: *Industrial Applications of fuzzy Control*, M. Sugeno Ed., Amsterdam: North Holland, 1985, p. 41-62.

- [STA,87] Stachowicz, M.S., Kochanska, M.E., "Fuzzy modeling of the Process", IFSA, Tokyo Japon, Juillet 1987, p. 86-89.
- [TAK,85] Takagi, T., Sugeno, M., "Fuzzy Identification of Systems and its Applications to Modeling and Control", IEEE Transaction on Systems, Man and Cybernetics, vol. SNC-15 N° 1, Janvier/Février 1985, p. 116-132.
- [THE,89] Thelliez, S., Vergé, M., Jaume, D., "Applications du formalisme d'état à la commande des systèmes continus", Ed.: Paris Eyrolles, 1989, p. 19-43.
- [TZA,90] Tzafestas, S., Papanikolopoulos, N., "Incremental fuzzy expert P.I.D. Control", IEEE Transaction on *Industrial Electronics*, Vol. 37 n° 5, Octobre 1990, p. 365-371.
- [VER,92] Verbruggen, H.B., Krijgsman, A.J., Bruijn, P.M., "Towards intelligent Control: Integration of AI in Control", dans: *Application of Artificial Intelligence in Process Control*, édité par: L. Boullart, A. Krijgsman et R.A. Vingerhoeds, Pergamon Press New York, 1992, p. 223-249.
- [VIL,92] de Villier, J., Barnard, E., "Backpropagation neural Nets with one and two hidden Layers", IEEE Transactions on *Neural Networks*, Vol. 4 n° 1, Janvier 1992, p.136-141.
- [YAM,89] Yamakawa, T., "Stabilization of an inverted Pendulum by a high-speed fuzzy Logic Controller hardware System", Elsevier Science Publ. B.V., 89, p. 161-180.
- [ZAD,65] Zadeh, L.A., "Fuzzy Sets", *Information Control*, vol. 8, 1965, p. 338-353.
- [ZAD,92] Zadeh, L., "Soft Computing, fuzzy Logic and Neural Network", *La Lettre du Club Crin*, numéro spécial *Logique floue*, Mai 1992, p. 2-3.

Annexe

A1- Equivalence exacte entre des lois de régulation floue et PI analytique

(Régulateur flou de type *Sugeno* avec l'utilisation du produit algébrique pour traduire le ET et de la somme bornée pour l'agrégation).

Les étapes d'une inférence sont détaillées dans le cas où quatre règles sont activées avec deux des quatre conclusions identiques (domaine de fonctionnement non-saturé).

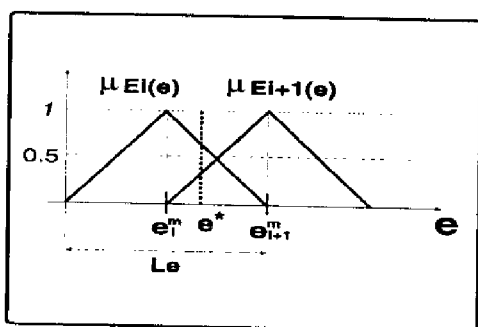


Figure 7.1 : Notations pour les EF.

Les EF des antécédents des règles considérées sont E_i , E_{i+1} et ΔE_j , ΔE_{j+1} respectivement pour les entrées numériques e^* et Δe^* . Les degrés d'appartenance des entrées aux EF précédents sont exprimés dans les formules (7.1), la figure (7.1) illustrant ces relations.

$$\begin{cases} \mu_{E_i}(e^*) &= \frac{2(e^m_{i+1} - e^*)}{L_e} \\ \mu_{E_{i+1}}(e^*) &= 1 - \mu_{E_i}(e^*) \end{cases} \quad (7.1)$$

$$\{ e^* \in [e^m_i, e^m_{i+1}]; i \in [1, 6] \}$$

La formule générale de l'inférence (7.2) est utilisée avec la règle compositionnelle d'inférence de *Zadeh*, la conjonction de *Mamdani*, l'interprétation du *et* avec le *produit algébrique*, de l'agrégation avec la *somme bornée* et une défuzzification par une moyenne pondérée.

$$\left\{ \begin{array}{l} \forall e \in E; \forall \Delta e \in \Delta E; \Delta u \in \Delta U; l = i, i+1; m = j, j+1 : \\ \mu^*(\Delta u) = \underbrace{OU}_{R_{l,m}} \left\{ \underbrace{Sup}_{(e, \Delta e)} \text{MIN} \{ \mu_{E_l}(e), \mu_{\Delta E_m}(\Delta e), I[ET(\mu_{E_l}(e), \mu_{\Delta E_m}(\Delta e)), \mu_{\Delta U_{l+m}}(\Delta u)] \} \right\} \end{array} \right. \quad (7.2)$$

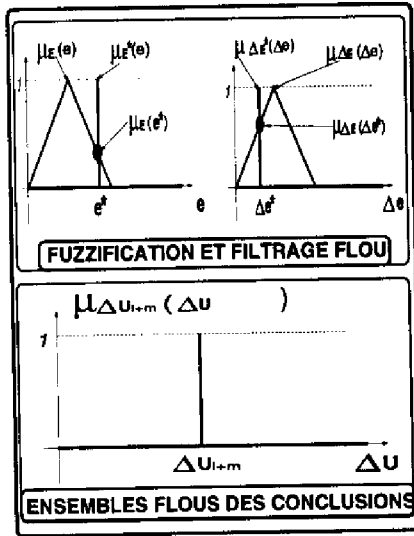


Figure 7.2 : EF codant les entrées et les sorties du RF.

La première étape de l'inférence consiste à calculer le résultat de la combinaison des entrées avec chacune des quatre règles $R_{l,m}$; $l = i, i+1$; $m = j, j+1$.

La première sous-étape est l'application de l'implication floue selon la formule (7.3). Les sorties étant codées sous forme de singletons (voir la figure (7.2)) et sachant que les degrés d'appartenance sont inférieurs ou égaux à 1, l'interprétation logique de l'implication est donnée en (7.4).

$$I = \text{MIN}[(\mu_{E_l}(e) \cdot \mu_{\Delta E_m}(\Delta e)), \mu_{\Delta U_{l+m}}(\Delta u)] \quad (7.3)$$

$$\begin{cases} \forall e \in E ; \forall \Delta e \in \Delta E : \\ I = \mu_{E_l}(e) \cdot \mu_{\Delta E_m}(\Delta e) & \text{si } \Delta u = \Delta u_{l+m} \\ I = 0 & \text{sinon} \end{cases} \quad (7.4)$$

La deuxième sous-étape est l'inférence pour une règle selon la formule (7.5). Les entrées étant fuzzifiées sous forme de singletons, voir la figure (7.2), le résultat est donné en (7.6).

$$\begin{cases} \text{pour } \Delta u = \Delta u_{l+m} : \\ \mu_{R_{l,m}}(\Delta u) = \underset{(e, \Delta e)}{\text{Sup}} \text{MIN} \{ \mu_{E'}(e), \mu_{\Delta E'}(\Delta e), [\mu_{E_l}(e) \cdot \mu_{\Delta E_m}(\Delta e)] \} \end{cases} \quad (7.5)$$

$$\begin{cases} \mu_{R_{l,m}}(\Delta u) = \mu_{E_l}(e^*) \cdot \mu_{\Delta E_m}(\Delta e^*) & \text{pour } \Delta u = \Delta u_{l+m} \\ \mu_{R_{l,m}}(\Delta u) = 0 & \text{sinon} \end{cases} \quad (7.6)$$

La deuxième étape de l'inférence consiste à calculer le résultat de la combinaison des entrées avec l'ensemble des quatre règles $R_{l,m}$; $l = i, i+1$; $m = j, j+1$. La première sous-étape est l'agrégation des résultats obtenus par chacune des règles selon la formule (7.7). L'opérateur d'agrégation *somme bornée* consiste à accorder une importance plus grande aux zones de recouvrement qu'avec le *maximum* traditionnellement utilisé. Dans le cas d'un même singleton inféré par deux règles différentes, la somme des deux degrés d'appartenance déduits est considérée. Le résultat de l'agrégation est donné en (7.8).

$$\begin{cases} \forall \Delta u \in \Delta U \\ \mu^*(\Delta u) = \underset{(l=i, i+1; m=j, j+1)}{\text{OU}} [\mu_{R_{l,m}}(\Delta u)] \end{cases} \quad (7.7)$$

$$\begin{cases} \mu^*(\Delta u_{i+j}) &= \mu_{R_{i,j}}(\Delta u_{i+j}) \\ \mu^*(\Delta u_{i+j+2}) &= \mu_{R_{i+1,j+1}}(\Delta u_{i+j+2}) \\ \mu^*(\Delta u_{i+j+1}) &= \text{MIN}[1, \mu_{R_{i,j+1}}(\Delta u_{i+j+1}) + \mu_{R_{i+1,j}}(\Delta u_{i+j+1})] \\ \mu^*(\Delta u_k) &= 0 \text{ pour } k \notin \{i+j, i+j+1, i+j+2\} \end{cases} \quad (7.8)$$

La deuxième sous-étape est la défuzzification selon la formule (7.9).

$$\Delta u^* = \frac{\sum_{l,m} \mu^*(\Delta u_{l+m}) \times \Delta u_{l+m}}{\sum_{l,m} \mu^*(\Delta u_{l+m})} \quad (7.9)$$

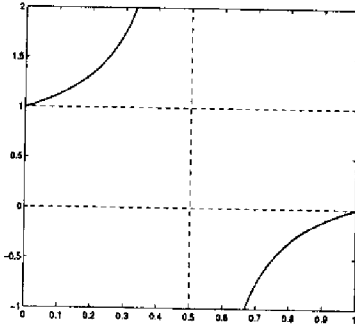


Figure 7.3 : $y = \frac{1-x}{1-2x}$.

A ce stade, il est possible de simplifier l'expression obtenue, sachant que l'équation (7.10) est vraie. En effet, il n'existe pas de valeurs de $\mu_{E_i}(e^*)$ et $\mu_{\Delta E_j}(\Delta e^*)$, comprises entre 0 et 1, telles que les relations (7.11) soient vérifiées, voir la courbe (7.3).

$$\mu_{R_{i,j+1}}(\Delta u_{i+j+1}) + \mu_{R_{i+1,j}}(\Delta u_{i+j+1}) \leq 1 \quad (7.10)$$

$$\begin{cases} \mu_{E_i}(e^*) > \frac{1-\mu_{\Delta E_j}(\Delta e^*)}{1-2\mu_{\Delta E_j}(\Delta e^*)} & \text{pour } \mu_{\Delta E_j}(\Delta e^*) < 0.5 \\ \mu_{E_i}(e^*) < \frac{1-\mu_{\Delta E_j}(\Delta e^*)}{1-2\mu_{\Delta E_j}(\Delta e^*)} & \text{pour } \mu_{\Delta E_j}(\Delta e^*) > 0.5 \end{cases} \quad (7.11)$$

En conséquence, l'expression (7.12) est celle du résultat final inféré; elle correspond à un système de *Sugeno*. Le dénominateur de cette expression vaut 1; ce qui permet d'exprimer plus simplement le résultat, voir l'équation (7.13).

$$\Delta u^* = \frac{\sum_{l,m} \mu_{R_{l,m}}(\Delta u_{l+m}) \times \Delta u_{l+m}}{\sum_{l,m} \mu_{R_{l,m}}(\Delta u_{l+m})} \quad (7.12)$$

$$\begin{aligned} \Delta u^* &= \Delta u_{i+j} \times [\mu_{E_i}(e^*) \cdot \mu_{\Delta E_j}(\Delta e^*)] + \Delta u_{i+j+2} \times [\mu_{E_{i+1}}(e^*) \cdot \mu_{\Delta E_{j+1}}(\Delta e^*)] + \dots \\ &\dots \Delta u_{i+j+1} \times [\mu_{E_i}(e^*) \cdot \mu_{\Delta E_{j+1}}(\Delta e^*) + \mu_{E_{i+1}}(e^*) \cdot \mu_{\Delta E_j}(\Delta e^*)] \end{aligned} \quad (7.13)$$

Le problème posé maintenant est de montrer que l'interpolation est linéaire quelque soient $e^* \in E$ et $\Delta e^* \in \Delta E$, sachant qu'il y a équivalence exacte aux valeurs modales. La démarche consiste donc à exprimer la valeur inférée, Δu^* , en fonction des entrées $(e^*, \Delta e^*)$ et des valeurs modales e^m_i et Δe^m_j . L'utilisation des équations (7.1) et de la définition de $L_{\Delta u}$, dans la formule (2.16), pour modifier l'expression (7.13), permet d'obtenir la formulation (7.14). L'expression des gains proportionnels et intégral équivalent en (2.18) et la relation d'équivalence aux valeurs modales (2.17) conduisent à l'équation d'une loi de régulation linéaire (7.15) pour tous les couples de l'espace d'entrée.

$$\Delta u^* = -2 L_{\Delta u} \times \left[\frac{e_{i+1}^m - e^*}{L_e} + \frac{\Delta e_{j+1}^m - \Delta e^*}{L_{\Delta e}} \right] + \Delta u_{i+j+2} \quad (7.14)$$

$$\Delta u^* = K_p e^* + K_I \Delta e^* \quad (7.15)$$

A2- Méthode de stabilité pour la supervision floue de régulateur PID

Il s'agit d'analyser la stabilité d'un système linéaire possédant des incertitudes structurées, variant dans le temps dans des domaines pré-déterminés.

1- Réécriture de la boucle fermée

La boucle de commande comprend le système du deuxième ordre (7.16), un régulateur PID académique de structure parallèle et un superviseur flou, voir la figure (7.4). Les sources de non-linéarités sont le système et les adaptations des gains du PID. Ces derniers sont modélisés par des paramètres incertains de la boucle de commande.

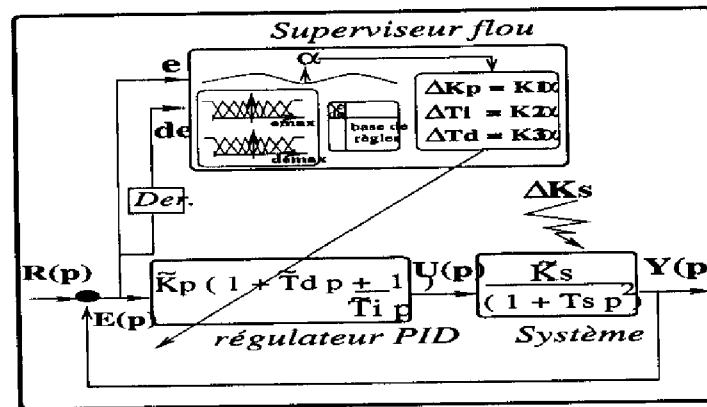


Figure 7.4 : Boucle de commande considérée.

Le système régulé est non linéaire à cause d'un gain \tilde{K}_s , variable en fonction de la référence, voir les formules (7.16) et (7.17).

$$G(p) = \frac{\tilde{K}_s}{(1 + T_s p)^2} \quad (7.16)$$

$R(k)$	75.5°	90°	100°	105°	115°
\tilde{K}_s	3.3	1.7	5.3	2.2	4.8

$$\tilde{K}_s = 0.00035 * R(k)^2 \quad (7.17)$$

Tableau 7.1 : Variations de \tilde{K}_s .

Le modèle de four est rendu linéaire par compensation adaptative de son gain (par une approche de type planification de gains). En effet, le gain proportionnel du PID est décomposé en deux termes: $\tilde{K}_p = K_p^* \times K_p$. K_p permet de réagir aux fluctuations de \tilde{K}_s à partir des mesures expérimentales du tableau (7.1) ou de la loi de perturbation (7.17), tel que $K_p \times \tilde{K}_s = K_f = 1$. Une autre approche aurait consisté à considérer \tilde{K}_s comme un autre paramètre incertain dans l'étude qui va suivre....

1.1- Problème posé: description de la boucle de régulation supervisée

La boucle fermée avec le régulateur PID peut être exprimée dans l'espace d'état par la description compagne (7.18).

$$\begin{cases} \dot{X} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -F & -E & -D \end{bmatrix} X + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} R \\ Y = \begin{bmatrix} J & H & G \end{bmatrix} X \\ D = \frac{2 T_s + K_f T_d \times K_p^*}{T_s^2} & E = \frac{1 + K_f \times K_p^*}{T_s^2} & F = \frac{K_f \times K_p^*}{T_i \times T_s^2} \\ G = J \times T_d \times T_i & H = J \times T_i & J = \frac{K_f \times K_p^*}{T_i \times T_s^2} \end{cases} \quad (7.18)$$

Pour pouvoir étudier la stabilité, nous allons reformuler le problème....

1.2- Nouvelle formulation du problème: Isolement des non-linéarités

La boucle de commande peut s'exprimer sous la forme (7.19). Les termes $\delta_i(t)$, $i = 1$ à 3, sont les incertitudes définies à partir des gains du PID.

$$\begin{cases} \dot{X} = \sum_i \{ \delta_i(t)^{[1 \times 1]} A_i^{[3 \times 3]} \} X^{[3 \times 1]} + A^{[3 \times 3]} X^{[3 \times 1]} \\ \delta_1(t) = \frac{K_p^*}{T_i} & \delta_2(t) = K_p^* & \delta_3(t) = K_p^* \times T_d \\ A_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -\frac{K_f}{T_s^2} & 0 & 0 \end{bmatrix} & A_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -\frac{K_f}{T_s^2} & 0 \end{bmatrix} \\ A_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\frac{K_f}{T_s^2} \end{bmatrix} & A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -\frac{1}{T_s^2} & -\frac{2}{T_s} \end{bmatrix} \end{cases} \quad (7.19)$$

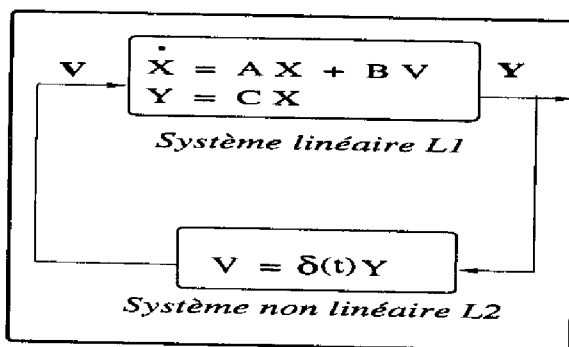


Figure 7.5 : Boucle fermée.

Les incertitudes peuvent être isolées dans une boucle de retour selon l'expression (7.20), voir

la figure (7.5). Cette transformation est nécessaire pour appliquer la condition de stabilité, dérivée du critère général des petits gains structurés: $gain(L_1) \times gain(L_2) < 1$.

$$\begin{cases} \dot{X} = AX + B^{[3 \times 3]} V^{[3 \times 1]} \\ Y = C^{[3 \times 3]} X \\ V = \delta(t)^{[3 \times 3]} Y^{[3 \times 1]} \end{cases} \quad (7.20)$$

La décomposition (7.20) est obtenue en utilisant les notations (7.21). C et B résolvant l'équation (7.22) sont recherchés. Il est possible de déduire la solution (7.23).

$$\begin{cases} \dot{X} = A X + B_1^{[3 \times 1]} V_1^{[1 \times 1]} + B_2^{[3 \times 1]} V_2^{[1 \times 1]} + B_3^{[3 \times 1]} V_3^{[1 \times 1]} \\ Y = \begin{pmatrix} C_1^{[1 \times 3]} \\ C_2^{[1 \times 3]} \\ C_3^{[1 \times 3]} \end{pmatrix} X \\ V = \begin{bmatrix} \delta_1(t) & 0 & 0 \\ 0 & \delta_2(t) & 0 \\ 0 & 0 & \delta_3(t) \end{bmatrix} Y^{[3 \times 1]} \end{cases} \quad (7.21)$$

$$\dot{X} = A X + \sum_i \{\delta_i(t) A_i\} X = A X + B \delta(t) C X = A X + \sum_i \{B_i \delta_i C_i\} X \quad (7.22)$$

$$\begin{cases} B = -\frac{K_t}{T_i^2} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \\ C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{cases} \quad (7.23)$$

2- Encadrement des incertitudes

2.1- Valeurs limites des gains du PID

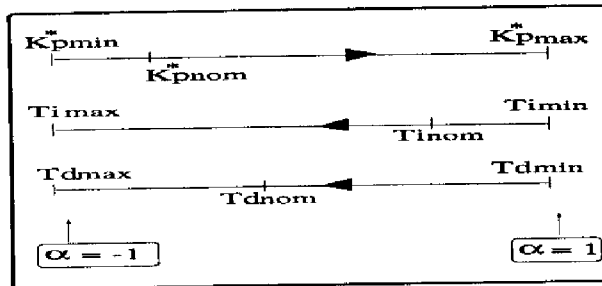


Figure 7.6 : Définition des intervalles de variation des gains du PID.

Les gains du PID sont définis par les équations (7.24), à partir du résultat de l'inférence α , des valeurs nominales des gains et des facteurs K_i , $i = 1$ à 3. Les intervalles de variation des gains sont choisis non symétriques autour de la valeur nominale, voir la figure (7.6).

$$\begin{cases} K_p^*(t) = K_p^*_{nom} + \alpha(t) \times K_1 \\ T_i(t) = T_{i,nom} - \alpha(t) \times K_2 \\ T_d(t) = T_{d,nom} - \alpha(t) \times K_3 \end{cases} \quad (7.24)$$

2.2- Valeurs limites des incertitudes

Les incertitudes s'exprime en fonction des gains du PID par les équations (7.25).

$$\delta_1(t) = \frac{K_p^*(t)}{T_i(t)} \quad \text{et} \quad \delta_2(t) = K_p^*(t) \quad \text{et} \quad \delta_3(t) = K_p^*(t) \times T_d(t) \quad (7.25)$$

Il s'agit de déterminer les bornes des incertitudes, en fonction des valeurs limites choisies pour les gains du PID. L'équation (7.26) fournit les bornes pour $\delta_1(t)$ et $\delta_2(t)$.

$$\frac{K_p^*_{min}}{T_{imax}} \leq \delta_1(t) \leq \frac{K_p^*_{max}}{T_{imin}} \quad \text{et} \quad K_p^*_{min} \leq \delta_2(t) \leq K_p^*_{max} \quad (7.26)$$

Dans le cas de $\delta_3(t)$, les bornes δ_{3min} et δ_{3max} doivent être calculées numériquement pour chaque application. Quelques indications peuvent toutefois être données. L'expression de $\delta_3(t)$ est donnée en (7.27).

$$\delta_3(t) = K_p^*(t) T_d(t) = [K_p^*_{nom} + \alpha(t)K_1] [T_{dnom} - \alpha(t)K_3] \quad (7.27)$$

Les relations (7.28), traduisant le fait que les gains du PID sont maximaux pour les valeurs limites de α inférées, permettent de déduire l'expression de $\delta_3(t)$ pour des valeurs positives et négatives de α , voir (7.29).

$$\begin{cases} K_p^*_{max} = K_p^*_{nom} + K_1 & \text{et} & T_{dmin} = T_{dnom} - K_3 & (\alpha = 1) \\ K_p^*_{min} = K_p^*_{nom} - K_1 & \text{et} & T_{dmax} = T_{dnom} + K_3 & (\alpha = -1) \end{cases} \quad (7.28)$$

$$\begin{cases} \alpha \geq 0 : \delta_3^1[\alpha(t)] = K_p^*_{nom} T_{dnom} + \alpha(t) [(K_p^*_{max} - K_p^*_{nom}) T_{dnom} - \dots \\ \dots K_p^*_{nom} (T_{dnom} - T_{dmin})] - \alpha(t)^2 [(K_p^*_{max} - K_p^*_{nom}) (T_{dnom} - T_{dmin})] \\ \alpha \leq 0 : \delta_3^2[\alpha(t)] = K_p^*_{nom} T_{dnom} + \alpha(t) [(T_{dnom} - T_{dmax}) K_p^*_{nom} - \dots \\ \dots T_{dnom} (K_p^*_{min} - K_p^*_{nom})] - \alpha(t)^2 [(T_{dnom} - T_{dmax}) (K_p^*_{min} - K_p^*_{nom})] \end{cases} \quad (7.29)$$

L'étude de la fonction $\delta_3(\alpha)$ permet d'obtenir des précisions sur les extréma de cette fonction. L'application des conditions de stationarité $-\delta_3^j(\alpha) = 0$, $j = 1, 2$ - et d'optimalité du deuxième ordre $-SIGNE\{\delta_3^j(\alpha)\}$, $j = 1, 2$ - conduit aux maxima $\delta_3^1(\alpha_1)$ et $\delta_3^2(\alpha_2)$, explicités dans les équations (7.30) et (7.31).

$$\begin{cases} \alpha_1 & = \frac{(K_p^*_{max} - K_p^*_{nom}) T_{dnom} - K_p^*_{nom} (T_{dnom} - T_{dmin})}{2 [(K_p^*_{max} - K_p^*_{nom}) (T_{dnom} - T_{dmin})]} \\ \delta_3^1(\alpha_1) & = K_p^*_{nom} T_{dnom} + \frac{[(K_p^*_{max} - K_p^*_{nom}) T_{dnom} - K_p^*_{nom} (T_{dnom} - T_{dmin})]^2}{4 (K_p^*_{max} - K_p^*_{nom}) (T_{dnom} - T_{dmin})} \end{cases} \quad (7.30)$$

$$\begin{cases} \alpha_2 &= \frac{(T_{dnom} - T_{dmax}) K_{p^* nom} - T_{dnom} (K_{p^* min} - K_{p^* nom})}{2 [(T_{dnom} - T_{dmax}) (K_{p^* min} - K_{p^* nom})]} \\ \delta_3^2(\alpha_2) &= K_{p^* nom} T_{dnom} + \frac{[(T_{dnom} - T_{dmax}) K_{p^* nom} - T_{dnom} (K_{p^* min} - K_{p^* nom})]^2}{4 (K_{p^* min} - K_{p^* nom}) (T_{dnom} - T_{dmax})} \end{cases} \quad (7.31)$$

Pour chaque problème posé, il faudra déterminer les bornes de $\delta_3(\alpha)$ à partir des calculs des maxima et des valeurs aux limites $\alpha = 1$ ou -1 .

3- Résolution de l'inégalité matricielle

3.1- Problème de stabilité posé

Il s'agit de trouver une fonction de *Lyapunov* L , définie à partir d'une matrice P symétrique et strictement positive, selon l'équation (7.32).

$$\begin{cases} \dot{X} &= A X + B V \\ L &= X^T P X \\ \dot{L} &= \dot{X}^T P X + X^T P \dot{X} \end{cases} \quad (7.32)$$

La condition de stabilité est exprimée en (7.33).

$$\begin{aligned} \dot{L} &= \begin{pmatrix} X \\ V \end{pmatrix} \underbrace{\begin{pmatrix} A^T P + PA & PB \\ B^T P & 0 \end{pmatrix}}_{T(P, \epsilon)} \begin{pmatrix} X \\ V \end{pmatrix} \leq -\epsilon \|X\|^2 \\ \Leftrightarrow &\begin{pmatrix} X \\ V \end{pmatrix} \begin{pmatrix} A^T P + PA + \epsilon I & PB \\ B^T P & 0 \end{pmatrix} \begin{pmatrix} X \\ V \end{pmatrix} \leq 0 \\ \Leftrightarrow &\begin{pmatrix} X \\ V_1 \\ V_2 \\ V_3 \end{pmatrix} \begin{pmatrix} A^T P + PA + \epsilon I & PB_1 & PB_2 & PB_3 \\ B_1^T P & 0 & 0 & 0 \\ B_2^T P & 0 & 0 & 0 \\ B_3^T P & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} X \\ V_1 \\ V_2 \\ V_3 \end{pmatrix} \leq 0 \end{aligned} \quad (7.33)$$

3.2- Formalisation des incertitudes

De l'équation (7.34), on déduit l'expression (7.35). On souhaite mettre les incertitudes sous la forme (7.36), ce qui conduit à l'identification (7.37).

$$\delta_i \in [\delta_{imin}, \delta_{imax}] \quad \text{et} \quad V_i = \delta_i(t) Y_i \quad , \quad i = 1..3 \quad (7.34)$$

$$\forall i \in [1, 3] : \quad (V_i - \delta_{imin} Y_i) (V_i - \delta_{imax} Y_i) \leq 0 \quad (7.35)$$

$$\forall i \in [1, 3] : \quad V_i^{2[1 \times 1]} Q_i^{[1 \times 1]} + 2 V_i S_i^{[1 \times 1]} Y_i^{[1 \times 1]} + R_i^{[1 \times 1]} Y_i^2 \geq 0 \quad (7.36)$$

$$Q_i = -1 \quad \text{et} \quad S_i = \frac{\delta_{i\min} + \delta_{i\max}}{2} \quad \text{et} \quad R_i = -\delta_{i\min} \times \delta_{i\max} \quad (7.37)$$

En utilisant les relations $Y_i = C_i X, \forall i = 1..3$, l'expression (7.36) devient (7.38); par exemple, dans le cas particulier où i vaut 1, on obtient (7.39).

$$\forall i : \quad V_i^2 Q_i + 2 V_i S_i C_i X + R_i X^T C_i^T C_i X \geq 0 \quad (7.38)$$

$$\begin{pmatrix} X \\ V_1 \\ V_2 \\ V_3 \end{pmatrix} \begin{pmatrix} \overbrace{C_1^T C_1 R_1}^{I_{n1}} & S_1 C_1 & 0 & 0 \\ S_1 C_1 & Q_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} X \\ V_1 \\ V_2 \\ V_3 \end{pmatrix} \geq 0 \quad (7.39)$$

3.4- Résolution du problème dit *S-procédure* par une approche *LMI*

Pour résoudre problème (7.33) sous la condition (7.38) (problème dit de *S-procédure*, [BOY,95]), il suffit de trouver des τ_i , positifs ou nuls, vérifiant l'équation (7.40).

$$T(P, \epsilon) + \sum_{i=1}^3 \tau_i I_{ni} \leq 0 \quad (7.40)$$

La résolution de ce problème se fait grâce aux procédures LMI (*Linear Matrix Inequalities*): il faut trouver P , ϵ et τ_i résolvant (7.40).

3.5- Conclusions de l'étude de stabilité

Si avec les domaines choisis $[\delta_{1\min}, \delta_{1\max}]$, $[\delta_{2\min}, \delta_{2\max}]$ et $[\delta_{3\min}, \delta_{3\max}]$, l'inéquation matricielle (7.40) est résolue, alors les paramètres K_i sont déduits du tableau (7.2). Si aucune solution n'est trouvée pour le problème de stabilité, alors il faut recommencer la procédure avec de nouveaux choix de domaines de variation des gains du PID.

	$\alpha \geq 0$	$\alpha \leq 0$
$K_1 =$	$K_p^* \max - K_p^* \min$	$K_p^* \max - K_p^* \min$
$K_2 =$	$T_{i\max} - T_{i\min}$	$T_{i\max} - T_{i\min}$
$K_3 =$	$T_{d\max} - T_{d\min}$	$T_{d\max} - T_{d\min}$

Tableau 7.2 : Dédution des valeurs de K_1 , K_2 et K_3 .

Techniques floues et neuronales pour des problèmes de régulation

Résumé:

Notre travail a porté principalement sur des aspects méthodologiques et sur des caractérisations expérimentales des techniques floues et neuronales, avec deux bancs d'essai: un pendule inversé et des simulateurs de fours industriels.

Dans une première étape, nous avons travaillé sur des régulateurs flous de la première génération. Nous avons proposé une méthode pour minimiser les erreurs dues à la quantification lors des mises en oeuvre avec table de décision. D'autre part, des études expérimentales ont permis d'expliquer l'amélioration de résultats de régulations floues par rapport à des régulations conventionnelles. Nous avons ainsi souligné les rôles, parfois bénéfiques, des non-linéarités liées aux interpolations ou aux saturations naturelles. Dans une deuxième étape, nous nous sommes intéressée aux structures de régulation neuronale directe, utilisant l'apprentissage spécialisé avec l'algorithme de rétropropagation. Nous avons développé une stratégie de régulation neuronale à régime glissant, automatiquement élaborée à partir d'une connaissance initiale très faible. Nous avons validé cette méthode avec des modèles simulés du pendule inversé et vérifié la possibilité de stabiliser le système réel. Dans une troisième étape, nous nous sommes penchée sur des améliorations de lois de régulation PID (couramment utilisées dans l'industrie), par introduction de non linéarités significatives. Dans un premier temps, nous avons proposé deux méthodologies de réglage d'adaptateurs flous de PID (parfois appelés *superviseurs flous*), agissant à partir de la position dans le plan de phase. Une des méthodes de supervision inclut des garanties de stabilité. Dans un deuxième temps, nous avons mis en oeuvre et amélioré une méthode d'auto-réglage d'un régulateur flou par un système flou, partant d'une loi de régulation floue initiale quasi-linéaire (résultant de l'application des règles d'équivalence). Des améliorations ont été apportées sur les fours par rapport à une régulation PID industrielle.

Mots Clefs: *Régulateur PID, Régulateur flou, Superviseur flou, Régulateur neuronal, Régulation adaptative, Auto-réglage, Stabilisation d'un pendule inversé, Régulation de température.*

Abstract:

Our work mainly focused on methodological aspects and experimental characterizations of fuzzy and neural techniques, with two benchmarks: an inverted pendulum and industrial furnaces.

In the first place, we have worked on fuzzy controllers of first generation. We have proposed a method for minimizing the errors due to quantization, in the case of look-up table implementations. Moreover, experimental studies have allowed to explain some improvements compared to some conventional control results. Thus, we have emphasized, in some cases, the good influence of the non-linearities due to interpolation and natural saturation. In the second place, we were interested in direct neural control structures, using specialized learning with the backpropagation procedure. We have developed a neural sliding-mode control strategy, automatically build from a very poor initial knowledge. We have validated this method with simulated inverted pendulum models and verified the possibility of stabilizing the real system. In the third place, we have improved PID control laws (widespread in the industry) by introducing significant non-linearities. Firstly, we have proposed two tuning methodologies of fuzzy PID adaptator (sometimes called *fuzzy supervisors*), acting from the phase-plane location. One of the method includes stability guarantees. Secondly, we have implemented and improved a fuzzy auto-tuning procedure of a fuzzy controller, starting from an initial quasi-linear control law (resulting of the equivalence principle use). Improvements have been brought on the furnaces compared to an industrial PID control.

Key Words: *PID Controller, fuzzy Controller, fuzzy Supervisor, neural Controller, adaptive Control, auto-tuning, stabilization of an inverted Pendulum, Furnace Control.*
