

T

THESE

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE

Pour obtenir le grade de

DOCTEUR D'UNIVERSITE

Spécialité Productique : Automatique et Informatique Industrielle

par

Laurent DUBOIS

Ingénieur IDN (Ecole Centrale de Lille)

UTILISATION DE LA LOGIQUE FLOUE DANS LA COMMANDE DES SYSTEMES COMPLEXES

Soutenu le 26 Septembre 1995 devant le jury d'examen

Président : G. DAUPHIN-TANGUY

Rapporteurs : A. EL MOUDNI
F. ROTELLA
D. WILLAEYS

Examineurs : P. BORNE
A.M. DESODT
P. MILLERAT
A. OUSTALOUP
J. ROZINOER



Travaux réalisés sous la direction du Professeur P. BORNE au Laboratoire d'Automatique et d'Informatique Industrielle de Lille (LAIL URA-CNRS D1440), Ecole Centrale de Lille, Villeneuve d'Ascq, France.

Avant-Propos

Le travail présenté dans ce mémoire a été effectué au Laboratoire d'Automatique et d'informatique Industrielle de Lille de l'Ecole Centrale de Lille (Unité de Recherche Associée au CNRS D-1440), sous la direction de Monsieur le Professeur Pierre BORNE, Directeur Scientifique de l'Ecole Centrale de Lille, Vice-Président d'IEEE/SMC et de l'IMACS.

Qu'il me soit permis ici de le remercier très vivement pour sa disponibilité malgré ses lourdes charges, la confiance qu'il m'a témoignée ainsi que pour l'enthousiasme qu'il a su me communiquer au cours de ces trois années.

Je tiens à remercier également Madame le Professeur Geneviève DAUPHIN-TANGUY, Directeur du LAIL, pour l'honneur qu'elle me fait en acceptant d'être le président de ce jury.

Je tiens à exprimer ma profonde gratitude au Professeur A. EL MOUDNI de l'ENIB qui a accepté d'être rapporteur de ce mémoire.

Je suis très honoré que Monsieur le Professeur F. ROTELLA de l'ENIT ait accepté d'être rapporteur de ce mémoire et je tiens à lui exprimer toute ma gratitude.

Je tiens également à assurer toute ma reconnaissance au Professeur D. WILLAEYS de l'Université de Valenciennes qui a accepté d'être rapporteur de ce mémoire.

Qu'il me soit permis de remercier Madame A.M. DESODT de l'Université de Lille I pour l'honneur qu'elle me fait en acceptant de participer au jury de cette thèse.

Je remercie également Monsieur P. MILLERAT d'avoir accepté d'être membre de ce jury de thèse.

Je tiens à exprimer toute ma reconnaissance au Professeur A. OUSTALOUP de l'ENSERB qui me fait l'honneur de participer à ce jury de thèse.

Je suis également très honoré de la présence de Monsieur A. ROZINOER, Directeur des études de l'Ecole Centrale de Lyon, qui a accepté d'être membre du jury de cette thèse et je tiens à le remercier.

Je tiens à exprimer toute ma reconnaissance à l'ensemble du personnel du LAIL pour l'ambiance cordiale et chaleureuse qu'il entretient au sein du laboratoire. De plus, je tiens à remercier "Max", imprimeur de l'Ecole Centrale de Lille, qui a effectué la reprographie de cette thèse dans les délais les plus brefs.

Qu'il me soit permis ici de remercier l'équipe "Bond-Graph Flou" (David, François, Janette, Jean-Marc, Moncef, Philippe, Sonia, Stéphane) pour son amitié au cours de ces années.

Table des matières

Introduction générale	9
Chapitre 1 : Notions générales.....	11
1 Introduction.....	13
1.1 Présentation de la logique floue	13
1.2 Historique	14
2 Définitions.....	15
2.1 Introduction	15
2.2 Sous-ensemble flou	15
2.3 Opérations sur les sous-ensembles flous	17
2.3.1 <i>Opérateur d'intersection</i>	18
2.3.2 <i>Opérateur d'union</i>	20
2.3.3 <i>Opérateur de négation</i>	21
2.3.4 <i>Projection d'un sous-ensemble flou</i>	22
2.4 Règle floue	23
2.5 Inférence floue	25
2.6 Agrégation de règles floues	27
3 Commande floue.....	28
3.1 Introduction	30
3.2 Commande floue	30
3.2.1 <i>Structure d'une commande floue</i>	30
3.2.2 <i>Base de règles et définitions</i>	30
3.2.3 <i>Interface de fuzzification</i>	33
3.2.3 <i>Mécanisme d'inférence</i>	35
3.2.4 <i>Interface de défuzzification</i>	35

3.3 Exemples d'application	39
3.3.1 <i>Commande floue d'un système du premier ordre</i>	39
3.3.2 <i>Commande floue du système du pendule inversé</i>	46
3.4 Conclusions	50
3.4.1 <i>Avantages des commandes floues</i>	50
3.4.2 <i>Inconvénients des commandes floues</i>	50
3.5 Objectifs fixés	51

Chapitre 2 : Optimisation des commandes

à base de règles floues.....53

1 Introduction.....	55
2 Historique.....	57
2.1 Système flou considéré	57
2.2 Détermination de la fonction f_S	59
2.3 Réalisation d'une commande floue adaptative	61
2.3.1 <i>Présentation</i>	61
2.3.2 <i>Remarques</i>	62
3 Analyse d'un système flou	63
3.1 Présentation	63
3.2 Fuzzification par la méthode du singleton	63
3.2.1 <i>Analyse</i>	63
3.2.2 <i>Commande floue adaptative</i>	65
3.3 Influence de la méthode de fuzzification	67
3.3.1 <i>Utilisation de la fonction min</i>	69
3.3.2 <i>Utilisation de la fonction produit</i>	71
3.3.3 <i>Remarques</i>	72
4 Conclusions.....	74

Chapitre 3 : COMMANDE FLOUE MULTI-MODELE77

1 Introduction.....	79
2 Principe de la commande multi-modèle.....	80
2.1 Utilisation d'un modèle linéarisé	80
2.2 Approche multi-modèle	80
3 Commande floue multi-modèle.....	82
3.1 Historique de la commande multi-modèle	82
3.1.1 <i>Modélisation floue de Takagi & Sugeno</i>	82
3.1.2 <i>Commande multi-modèle de Ishigame</i>	83
3.1.3 <i>Commande basée sur une modélisation floue de Johansen</i>	84
3.1.4 <i>Conclusions</i>	85
3.2 Approches considérées	86
3.2.1 <i>Système considéré</i>	86
3.2.2 <i>Modèles du système</i>	87
3.2.3 <i>Validité des modèles</i>	87
3.2.4 <i>Résumé</i>	87
4 Utilisation de modèles locaux.....	89
4.1 Introduction	89
4.2 Commande floue multi-modèle	90
4.2.1 <i>Structure</i>	90
4.2.2 <i>Principe de fonctionnement</i>	91
5 Exemples d'application utilisant des modèles locaux.....	102
5.1 Commande multi-modèle par mode glissant	102
5.1.1 <i>Commande par mode glissant</i>	102
5.1.2 <i>Commande multi-modèle</i>	105
5.1.3 <i>Simulations</i>	106

5.2 Commande sous-optimale multi-critère	110
5.2.1 <i>Position du problème</i>	110
5.2.2 <i>Commande sous-optimale d'un système du premier ordre</i>	110
5.2.3 <i>Simulations</i>	113
6 Utilisation de modèles génériques.....	115
6.1 Introduction	115
6.2 Commande floue multi-modèle utilisant des modèles génériques	116
6.2.1 <i>Structure</i>	116
6.2.2 <i>Principe de fonctionnement</i>	116
6.3 Exemple d'application utilisant des modèles génériques	118
6.3.1 <i>Présentation</i>	118
6.3.2 <i>Simulations</i>	119
7 Conclusions.....	121

Conclusion générale	123
Annexe.....	127
Annexe 1 : Démonstration du théorème de Wang	129
Annexe 2 : Démonstration du théorème d'approximation	133
Bibliographie	137

Introduction générale

Afin de répondre aux nouveaux défis de l'humanité, les ingénieurs sont amenés à développer des systèmes de plus en plus complexes. Cette complexité se traduit par le fait que l'on ne peut plus négliger des phénomènes jusqu'à présent considérés comme secondaires telle la flexion d'un bras de robot. Par conséquent, la représentation mathématique d'un tel système aboutit à la création d'un modèle généralement fortement non-linéaire et/ou incertain. Dans ce cas, la détermination d'une loi de commande performante peut être compromise à cause de problèmes dans la résolution d'équations algèbro-différentielles.

La théorie de la logique floue initialisée par L. A. Zadeh a pour principal intérêt de prendre en compte l'analyse qualitative d'un expert. Les commandes floues dérivées de ce principe n'utilisent alors aucun modèle explicite du système ; elles sont en effet basées sur l'intégration de la stratégie de commande d'un expert. On évite par conséquent les problèmes d'une modélisation complexe d'un système. Les commandes floues ont montré par le succès de leurs utilisations dans des applications industrielles leur utilité dans le contrôle des systèmes.

Néanmoins, la commande floue n'est pas exempte de problèmes dont les plus cruciaux sont le réglage des différents paramètres et la difficulté de dominer le comportement transitoire du système commandé.

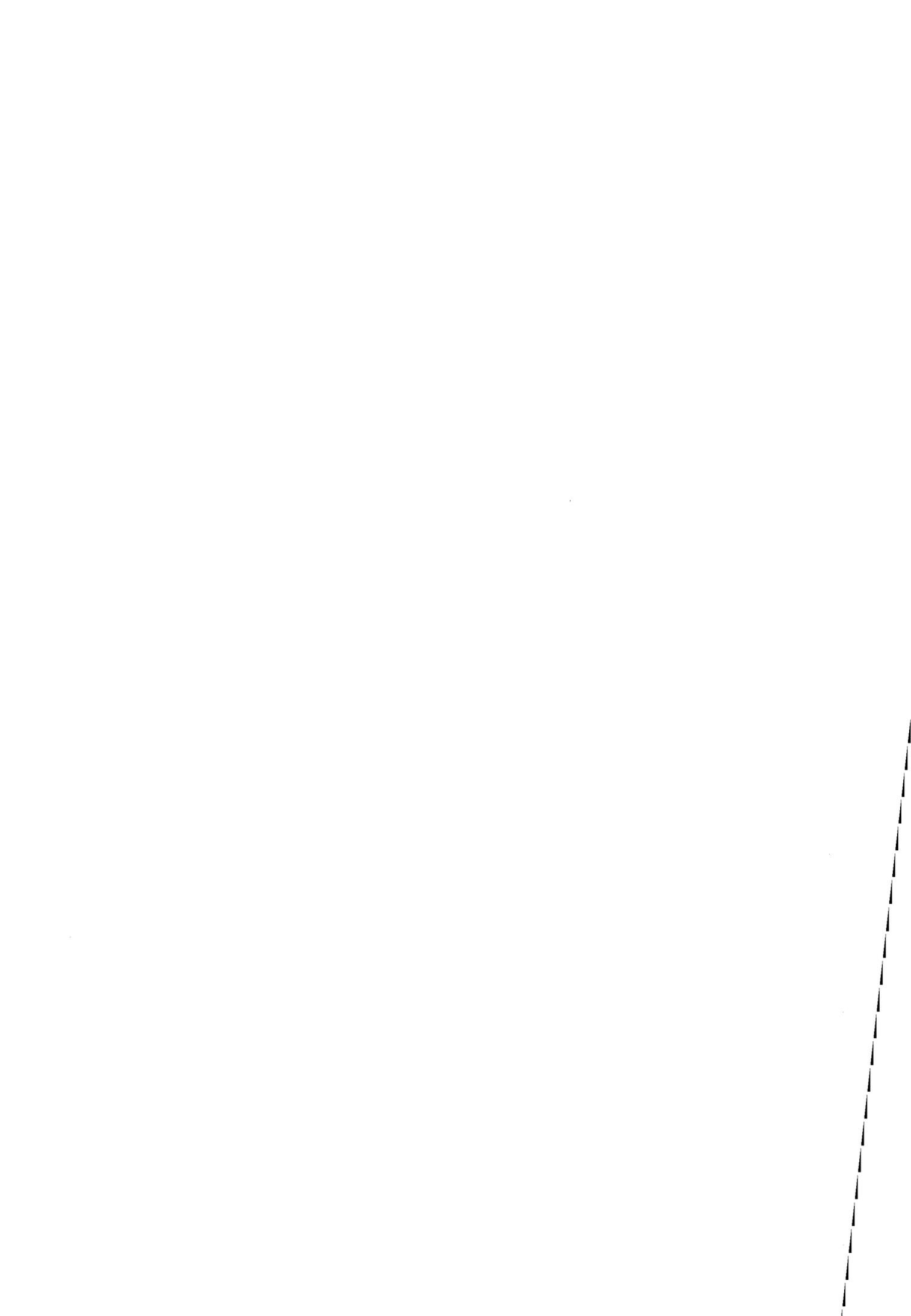
Nos travaux ont pour objectif de contribuer à résoudre ces deux problèmes, tout d'abord par l'analyse d'un système flou et ensuite par l'utilisation de la théorie des possibilités dérivée de la logique floue, dans une approche multi-modèle de la commande de systèmes complexes.

L'organisation de ce mémoire est la suivante. Dans le premier chapitre, nous rappellerons les principes de la logique floue et illustrerons par des exemples les principaux avantages et inconvénients de son application dans la commande de système.

L'analyse de la représentation non-linéaire d'un système flou et l'étude de l'influence de ses nombreux paramètres, et en particulier de la méthode de fuzzification choisie, sur les relations entrée-sortie feront l'objet du deuxième chapitre.

Enfin, dans le troisième chapitre, nous présenterons la commande multi-modèle d'un système complexe utilisant la théorie des possibilités afin de fusionner les différentes commandes proposées par chacun des modèles. Les avantages de la commande multi-modèle seront illustrés par des exemples d'application.

Chapitre 1 : NOTIONS GENERALES



1 Introduction

1.1 Présentation de la logique floue

La logique floue est l'une des approches envisagées afin d'accroître "l'intelligence" des machines, c'est-à-dire à analyser l'environnement extérieur et à s'adapter à celui-ci. Pour ce faire, ces approches tentent d'imiter le raisonnement humain. En effet, l'homme, pour parvenir à prendre une décision, analyse les informations peu précises ou incomplètes du monde extérieur en fonction de ses expériences antérieures sur des situations semblables (figure 1.1).

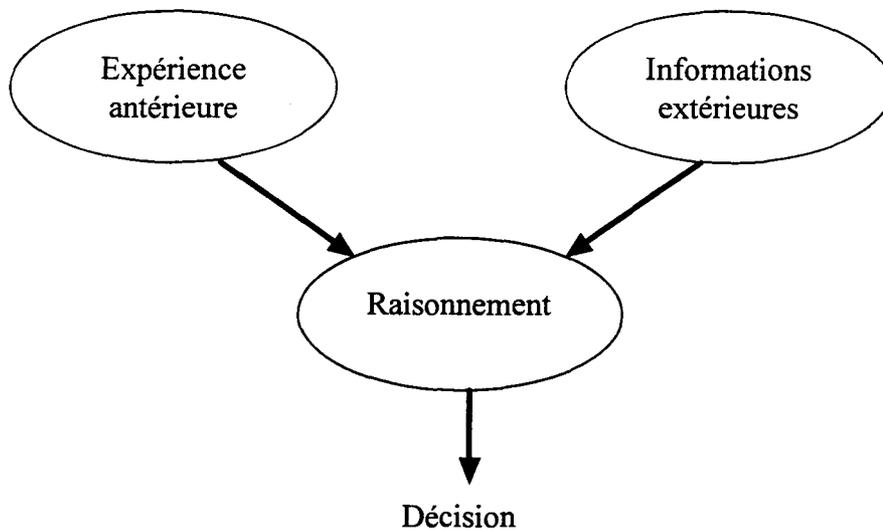


Figure 1.1 : Structure du raisonnement humain.

Le tableau 1.1 présente trois approches particulières utilisées afin d'approcher ce schéma de raisonnement : les réseaux de neurones, l'intelligence artificielle et la logique floue.

	Réseaux de Neurones	Intelligence Artificielle	Logique floue
Expérience	Mémoire distribuée	Base de règles	Base de règles
Informations extérieures	Couche de neurones en contact avec l'extérieur	Fait déterminé	Fait observé, non déterminé
Méthode de raisonnement	Interaction entre les neurones	Inférence	Inférence floue

Tableau 1.1 : Trois approches de l'intelligence artificielle.

1.2 Historique

Les premiers travaux sur l'étude des systèmes complexes par la logique floue sont l'oeuvre de Lotfi A. Zadeh ([Zad 1965] [Zad 1973]). Ces travaux concernaient la modélisation d'un système physique par un ensemble de règles ayant la forme de l'équation (1.1).

SI des conditions sont réalisées ALORS des conclusions peuvent être déduites. (1.1)

Dans ce type de règle, il est fait usage de termes linguistiques, tels que *grand* ou *chaud*, permettant de rendre compte de la capacité de l'être humain à raisonner avec des concepts imprécis. Par exemple, la règle suivante peut être utilisée afin de réguler la température d'un bain.

SI la température de l'eau est *élevée* ALORS ajouter de l'eau *froide*. (1.2)

Dans cette équation, les termes *élevée* et *froide* correspondent à des intervalles de température que l'utilisateur considère comme pouvant être associé à ceux-ci et non pas à des valeurs précises.

Par la suite, des chercheurs comme Mamdani [Man 1974] commencèrent à utiliser cette approche floue afin de commander des systèmes. Dans un premier temps, les commandes floues furent construites en s'inspirant des actions de l'opérateur humain capable de commander un système sans en connaître une définition rigoureuse.

Depuis, l'intérêt pour les commandes utilisant la logique floue n'a pas cessé de croître. Ce phénomène s'explique tant par la simplicité de mise en oeuvre de ce type de commande que par son application très rapide pour des systèmes industriels.

La structure de ce chapitre est la suivante. Dans un premier temps, on présentera les définitions utilisées en logique floue. Ensuite, l'application de cette théorie à la commande de système sera exposée. Enfin deux exemples de réalisation de commande floue seront étudiés afin de montrer les avantages et les inconvénients de ce type de commande.

2 Définitions

2.1 Introduction

La logique floue peut être envisagée comme une extension de la logique booléenne. Ainsi, alors qu'en logique classique l'appartenance d'un élément à un sous-ensemble donné est une valeur binaire (0 ou 1), la théorie des sous-ensembles flous considère cette appartenance comme une valeur réelle dans l'intervalle $[0 ; 1]$. Comme dans le cas de la logique conventionnelle, une valeur nulle implique la non-appartenance de l'élément au sous-ensemble, tandis que la valeur 1 indique que l'élément appartient au sous-ensemble considéré.

2.2 Sous-ensemble flou

Le sous-ensemble flou est l'élément de base de la logique floue. Il est caractérisé par sa fonction d'appartenance, fonction continue et à valeurs dans l'intervalle $[0 ; 1]$. Celle-ci définit le degré d'appartenance d'un élément au sous-ensemble flou considéré.

Définition 1.1 :

Un sous-ensemble flou F d'un univers de discours U est défini par :

$$F = \{x \in U; \mu_F(x)\}. \quad (1.3)$$

Définition 1.2 :

La fonction d'appartenance μ_F , caractérisant le sous-ensemble flou F de l'univers de discours U , est définie par :

$$\begin{aligned} \mu_F: U &\rightarrow [0 ; 1] \\ x &\mapsto \mu_F(x). \end{aligned} \quad (1.4)$$

L'univers de discours U (appelé de la sorte car à l'origine les sous-ensembles flous étaient relatifs à des termes linguistiques) est tel que tous les éléments considérés appartiennent de façon certaine à U . Par conséquent, la fonction d'appartenance caractérisant un univers de discours U est :

$$\forall x \in U, \mu_U(x) = 1. \tag{1.5}$$

Les univers de discours représentent les domaines d'évolution des variables. Par exemple, si l'on considère une vanne dont l'ouverture est limitée à 80%, l'univers de discours correspondant peut être défini par l'intervalle $[0 ; 0,8]$ et on peut envisager, sur cet univers de discours, le sous-ensemble flou *Ouverture moyenne* caractérisé par la fonction d'appartenance :

$$\begin{aligned} \mu_{\text{Ouverture moyenne}} : [0 ; 0,8] &\rightarrow [0 ; 1] \\ x &\mapsto \mu_{\text{Ouverture moyenne}}(x). \end{aligned} \tag{1.6}$$

Une représentation graphique possible est donnée sur la figure suivante :

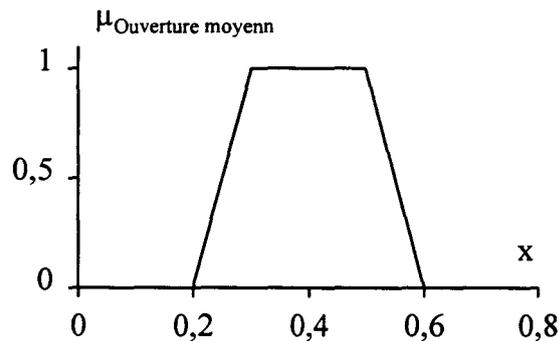


Figure 1.2 : Représentation graphique du sous-ensemble flou "Ouverture moyenne".

Les formes des fonctions d'appartenance usuelles permettant de décrire la confiance dans des faits de la forme *x est F* sont données figure 1.3.

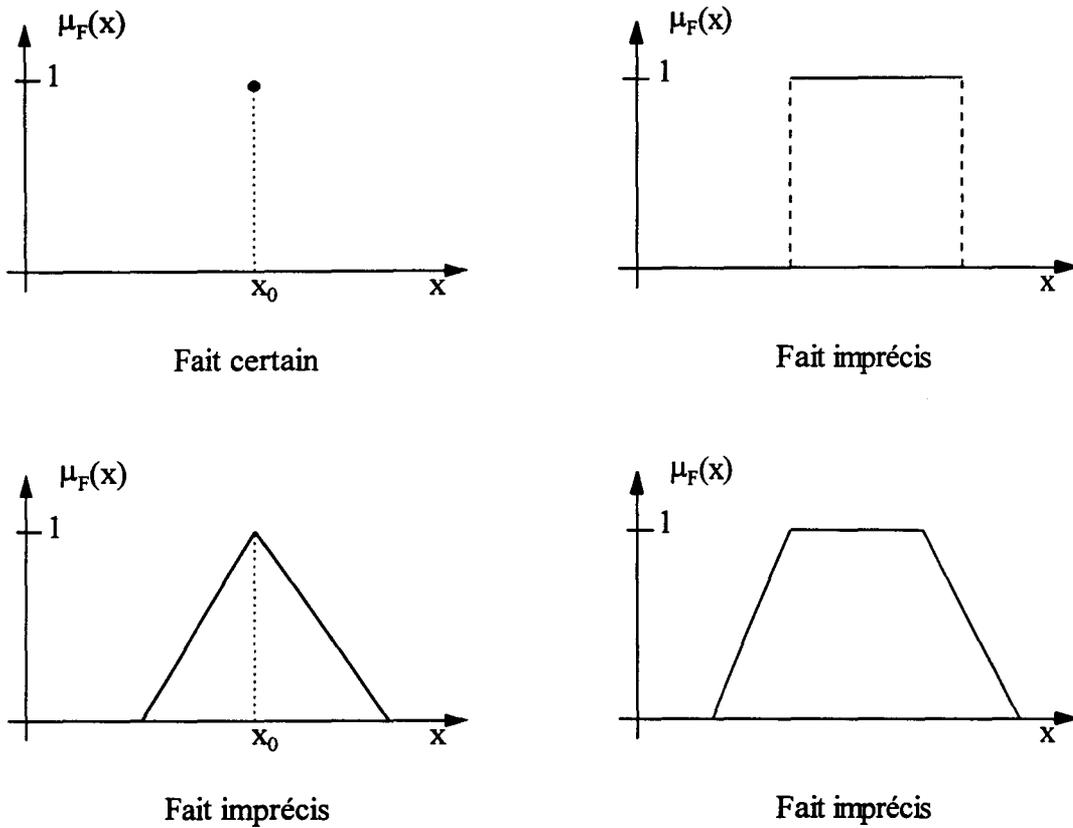


Figure 1.3 : Différentes représentations du fait x est F .

2.3 Opérations sur les sous-ensembles flous

Les opérations sur des sous-ensembles flous génèrent de nouveaux sous-ensembles flous caractérisés par des fonctions d'appartenance. Par conséquent, une opération sur des sous-ensembles flous est réalisée par l'intermédiaire des fonctions d'appartenance les caractérisant. Les définitions qui suivent sont tirées de l'article de S. Weber [Web 1983].

2.3.1 Opérateur d'intersection

Définition 1.3 :

Une fonction \wedge est un opérateur d'intersection si et seulement si :

$$\begin{aligned} & \wedge : [0 ; 1] \times [0 ; 1] \rightarrow [0 ; 1], \\ & \wedge \text{ est non décroissante pour chaque argument,} \\ & \wedge \text{ est commutative,} \\ & \wedge \text{ est associative,} \\ & \wedge \text{ a 1 pour élément neutre.} \end{aligned} \tag{1.7}$$

Les fonctions possédant ces propriétés sont appelées des normes triangulaires (T-norme).

Ainsi, l'intersection de deux sous-ensembles flous A et B, d'un même univers de discours U et définis par leurs fonctions d'appartenance $\mu_A(x)$ et $\mu_B(x)$, est caractérisée par la fonction d'appartenance $\mu_{A \cap B}$ définie par :

$$\begin{aligned} \mu_{A \cap B} : U & \rightarrow [0 ; 1] \\ x & \mapsto \mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x). \end{aligned} \tag{1.8}$$

Les T-normes sont aussi utilisées pour définir le produit cartésien. Ainsi, pour deux sous-ensembles flous A et B des univers de discours U et V le produit cartésien $A \times B$ est caractérisé par la fonction d'appartenance $\mu_{A \times B}$:

$$\begin{aligned} \mu_{A \times B} : U \times V & \rightarrow [0 ; 1] \\ (x, y) & \mapsto \mu_{A \times B}(x, y) = \mu_A(x) \wedge \mu_B(y). \end{aligned} \tag{1.9}$$

La figure qui suit présente deux T-normes parmi les plus utilisées en logique floue (fonctions *min* et *produit*).

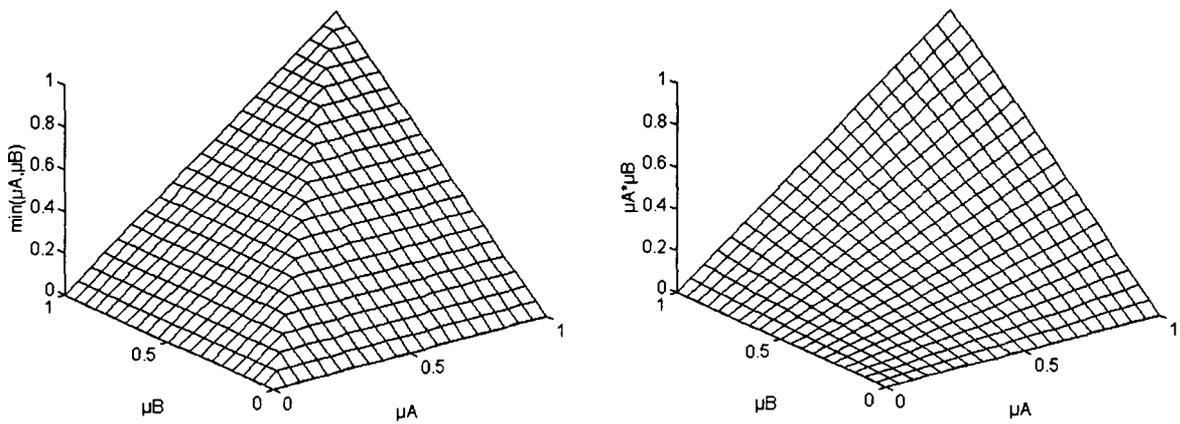


Figure 1.4 : Différents opérateurs d'intersection \wedge .

On retrouve pour les couples $\{\mu_A; \mu_B\}$ extrêmes (i.e. $\{0; 0\}$, $\{0; 1\}$, $\{1; 0\}$, $\{1; 1\}$) les valeurs de la table de vérité du ET logique données dans le tableau 1.2.

	B	0	1
A			
0		0	0
1		0	1

Tableau 1.2 : Table de vérité du ET logique.

Remarque :

La fonction *min* majore toutes les T-normes. Ainsi,

$$\forall (a, b) \in [0 ; 1]^2, \forall \wedge \in \{T - \text{normes}\}; \min(a, b) \geq a \wedge b. \quad (1.10)$$

2.3.2 Opérateur d'union

Définition 1.4 :

Une fonction \vee est un opérateur d'union si et seulement si :

$$\begin{aligned} &\vee : [0 ; 1] \times [0 ; 1] \rightarrow [0 ; 1], \\ &\vee \text{ est non décroissante pour chaque argument,} \\ &\vee \text{ est commutative,} \\ &\vee \text{ est associative,} \\ &\vee \text{ a } 0 \text{ pour élément neutre.} \end{aligned} \tag{1.11}$$

Les fonctions possédant ces propriétés sont appelées des conormes triangulaires (T-conorme).

Ainsi, l'union de deux sous-ensembles flous A et B d'un même univers de discours U, définis par leurs fonctions d'appartenance $\mu_A(x)$ et $\mu_B(x)$, est caractérisée par la fonction d'appartenance $\mu_{A \cup B}$:

$$\begin{aligned} \mu_{A \cup B} : U &\rightarrow [0 ; 1] \\ x &\mapsto \mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x). \end{aligned} \tag{1.12}$$

La figure suivante présente les opérateurs d'union les plus utilisées en logique floue.

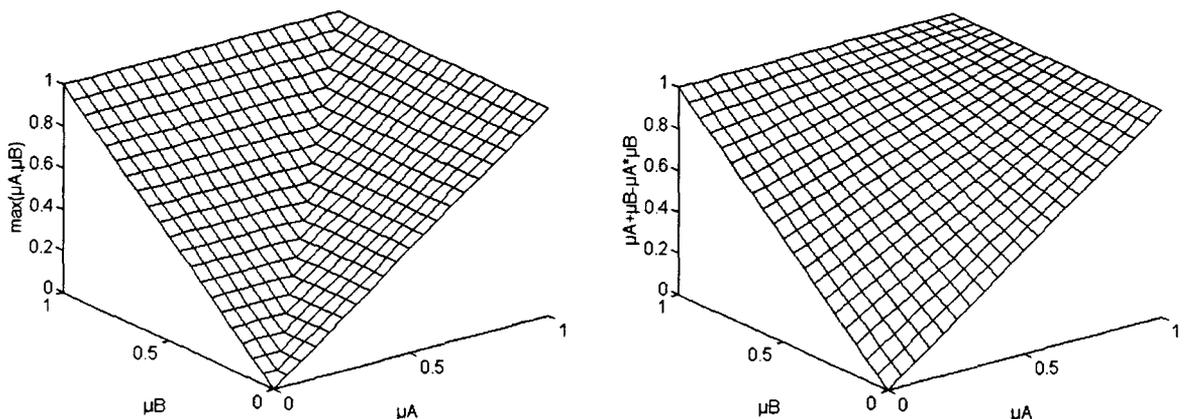


Figure 1.5 : Différents opérateurs d'union \vee .

Comme pour les T-normes, on retrouve pour les couples $\{\mu_A; \mu_B\}$ extrêmes les valeurs de la table de vérité du OU logique données dans le tableau 1.3.

	B	0	1
A			
0		0	1
1		1	1

Tableau 1.3 : Table de vérité du OU logique.

Remarque 1 :

La fonction max minore toutes les T-conormes. Ainsi,

$$\forall (a, b) \in [0 ; 1]^2, \forall v \in \{T\text{-conormes}\}; \max(a, b) \leq a \vee b. \quad (1.13)$$

Remarque 2 :

On peut observer que les opérateurs d'union et d'intersection ne sont en fait que des fonctions d'interpolation permettant de relier les valeurs obtenues par la logique booléenne. Par cet intermédiaire, on peut étendre les définitions de la logique classique à des valeurs réelles appartenant à l'intervalle [0 ; 1].

2.3.3 Opérateur de négation

Définition 1.5 :

Une fonction \neg est un opérateur de négation si et seulement si :

$$\begin{aligned} &\neg : [0 ; 1] \rightarrow [0 ; 1], \\ &\neg \text{ est continue,} \\ &\neg \text{ non croissante,} \\ &\neg(0) = 1 \text{ et } \neg(1) = 0, \\ &\text{de plus, } \neg \text{ est une négation stricte ssi } \forall a \neg(\neg(a)) = a. \end{aligned} \quad (1.14)$$

Un exemple d'opérateur de négation utilisé en logique floue est donné par l'expression :

$$\neg(a) = 1-a. \quad (1.15)$$

La figure suivante présente un exemple d'application de cet opérateur.

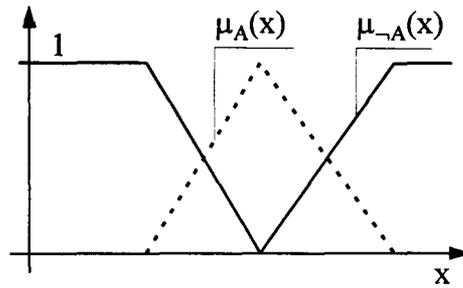


Figure 1.6 : Exemple d'application de l'opérateur de négation (1.15).

A partir de la définition de l'opérateur de négation, on introduit la notion d'opérateurs duaux.

Définition 1.6 :

Deux opérateurs \wedge et \vee sont dits duaux si et seulement si ils vérifient :

$$A \vee B = \neg(\neg(A) \wedge \neg(B)),$$

$$A \wedge B = \neg(\neg(A) \vee \neg(B)).$$

(1.16)

Le tableau 1.4 présente des opérateurs duaux pour l'opérateur de négation défini dans l'équation (1.15).

Opérateur d'intersection \wedge	Opérateur d'union \vee
$\min(\mu_A, \mu_B)$	$\max(\mu_A, \mu_B)$
$\mu_A \mu_B$	$\mu_A + \mu_B - \mu_A \mu_B$

Tableau 1.4 : Exemples d'opérateurs duaux.

2.3.4 Projection d'un sous-ensemble flou

Considérons un sous-ensemble flou A d'un univers de discours $U \times V$, caractérisé par sa fonction d'appartenance $\mu_A(x,y)$. On désire établir l'influence de ce sous-ensemble sur un univers de discours particulier U ou V. Pour ce faire, on utilise un opérateur de projection.

Définition 1.7 :

La projection d'un sous-ensemble flou A d'un univers de discours $U \times V$, caractérisé par sa fonction d'appartenance $\mu_A(x,y)$, sur U est le sous-ensemble flou $P_U(A)$ de U caractérisé par la fonction d'appartenance :

$$\mu_{P_U(A)}(x) = \sup_y (\mu_A(x,y)). \quad (1.17)$$

2.4 Règle floue

Une règle floue est la transcription de la relation qui lie deux (ou plus) sous-ensembles flous. Cette relation prend généralement la forme suivante :

$$\text{SI } x \text{ est } A \text{ ALORS } y \text{ est } B, \quad (1.18)$$

où A et B sont deux sous-ensembles flous des univers de discours U et V. Les faits *x est A* et *y est B* sont caractérisés par les fonctions d'appartenance $\mu_A(x)$ et $\mu_B(y)$.

Définition 1.8 :

Une règle floue $R = A \rightarrow B$, $A \subset U$ et $B \subset V$, est un sous-ensemble flou de l'univers de discours $U \times V$ caractérisé par la fonction d'appartenance μ_R définie à partir des fonctions d'appartenance $\mu_A(x)$ et $\mu_B(y)$ définissant les sous-ensembles flous A et B :

$$\begin{aligned} \mu_R : U \times V &\rightarrow [0 ; 1] \\ (x,y) &\mapsto \mu_R(x,y) = \mu_{A \rightarrow B}(x,y) = f(\mu_A(x), \mu_B(y)). \end{aligned} \quad (1.19)$$

L'implication floue caractérisant une règle floue de la forme $R = A \rightarrow B$ peut être définie comme une extension de la définition de l'implication *A implique B* utilisée en logique booléenne. Une définition de celle-ci est :

$$A \rightarrow B = (\neg A) \text{ OU } B, \quad (1.20)$$

dont la table de vérité est donnée dans le tableau suivant.

	B	0	1
A			
0		1	1
1		0	1

Tableau 1.5 : Table de vérité de l'implication logique.

Ainsi, la définition générale de l'implication floue est :

Définition 1.9 :

La fonction d'appartenance μ_R de la règle floue $R = A \rightarrow B$, $A \subset U$ et $B \subset V$ est :

$$\begin{aligned} \mu_R : U \times V &\rightarrow [0 ; 1] \\ (x, y) &\mapsto \neg(\mu_A(x)) \vee \mu_B(y). \end{aligned} \tag{1.21}$$

Dans cette équation, \neg et \vee sont, respectivement, des opérateurs de négation et d'union. Par exemple, en choisissant la fonction *max* comme l'opérateur d'union et l'opérateur de négation défini dans l'équation (1.15), on obtient :

$$\begin{aligned} \mu_R : U \times V &\rightarrow [0 ; 1] \\ (x, y) &\mapsto \max(1 - \mu_A(x), \mu_B(y)). \end{aligned} \tag{1.22}$$

Pourtant, pour l'ensemble des applications de la logique floue en automatique (modélisation et/ou commande), on ne considère que des systèmes passifs, c'est-à-dire des systèmes pour lesquels à une entrée nulle correspond une sortie nulle. La table de vérité de ce type de système est la suivante :

	B	0	1
A			
0		0	0
1		0	1

Tableau 1.6 : Table de vérité d'un système passif (A et B sont respectivement l'entrée et la sortie du système)

Par conséquent, afin de représenter de tels systèmes, les implications floues utilisées en automatique sont des T-normes (c'est-à-dire des opérateurs d'intersection \wedge). Ainsi, la règle floue R modélisant un système passif est caractérisée par la fonction d'appartenance μ_R :

$$\begin{aligned} \mu_R : U \times V &\rightarrow [0 ; 1] \\ (x, y) &\mapsto \mu_R(x, y) = \mu_A(x) \wedge \mu_B(y). \end{aligned} \tag{1.23}$$

Par exemple, si l'opérateur d'intersection choisi est la fonction *min*, on obtient :

$$\begin{aligned} \mu_R : U \times V &\rightarrow [0 ; 1] \\ (x, y) &\mapsto \mu_R(x, y) = \min(\mu_A(x), \mu_B(y)). \end{aligned} \tag{1.24}$$

2.5 Inférence floue

Une règle floue du type *SI x est A ALORS y est B* détermine la relation liant des sous-ensembles flous parfaitement définis A et B (i.e. des sous-ensembles dont on connaît les fonctions d'appartenance). Peut-on, en utilisant l'implication floue caractérisant cette règle, trouver le sous-ensemble flou B' correspondant à un fait *x est A'* où A' est un sous-ensemble flou différent de A ?

Le calcul de la fonction d'appartenance de B' est réalisé par l'intermédiaire du mécanisme d'inférence. Ce dernier tend à résoudre le problème si $A \rightarrow B$ alors $A' \rightarrow ?$. La description de ce problème est :

$$\begin{aligned}
 &\text{Fait flou: } x \text{ est } A', \\
 &\text{R\`egle: SI } x \text{ est } A \text{ ALORS } y \text{ est } B, \\
 &\text{Conclusion: } y \text{ est } B'.
 \end{aligned}
 \tag{1.25}$$

On veut trouver le sous-ensemble flou B' de l'univers de discours V , caract\eris\ee par sa fonction d'appartenance $\mu_{B'}$, \`a partir de la connaissance du sous-ensemble flou A' de l'univers de discours U (caract\eris\ee par $\mu_{A'}$) et de l'implication floue μ_R d\efinie sur $U \times V$.

D\efinition 1.10 :

Le sous-ensemble flou B' de l'univers de discours V solution du probl\eme (1.25) est la projection sur V du produit cart\esien entre le sous-ensemble flou A' et la relation floue R .

$$B' = P_V(A' \times R). \tag{1.26}$$

En reprenant les d\efinitions du produit cart\esien et de la projection floue, la fonction d'appartenance $\mu_{B'}(y)$ caract\erisant la solution B' est :

$$\mu_{B'}(y) = \sup_x (\mu_{A'}(x) \wedge \mu_R(x, y)), \tag{1.27}$$

o\`u l'op\erateur \wedge est une T-norme (e.g. la fonction *min*).

Exemple :

Consid\erons les fonctions d'appartenance des sous-ensembles flous A , A' et B (connus) pr\esent\ees sur la figure suivante.

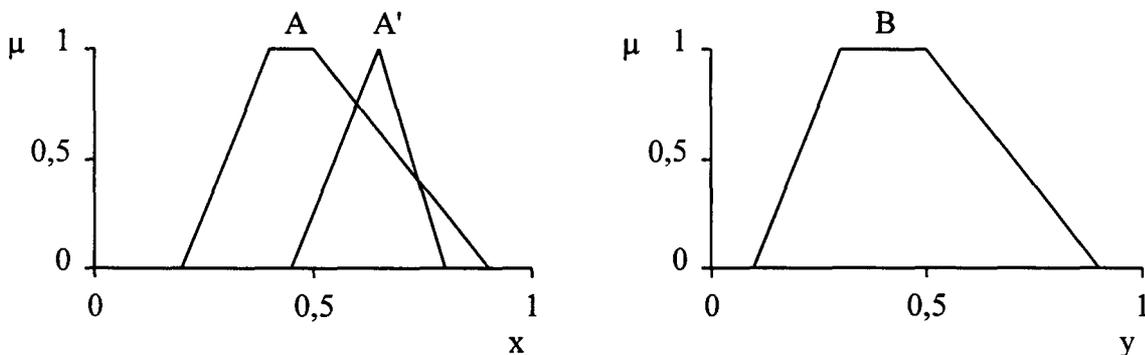


Figure 1.7 : Repr\esentation des fonctions d'appartenance des sous-ensembles flous A , A' et B .

Le tableau qui suit pr\esente le choix des op\erateurs utilis\es dans le calcul du sous-ensemble flou B' .

Opérateur	Choix
Produit cartésien	min
Implication floue	min

Tableau 1.7 : Choix des opérateurs.

La fonction d'appartenance du sous-ensemble flou B' est alors :

$$\begin{aligned}
 \mu_{B'}(y) &= \sup_x (\mu_{A'}(x) \wedge \mu_R(x, y)) \\
 &= \sup_x (\min(\mu_{A'}(x), \min(\mu_A(x), \mu_B(y)))) \\
 &= \min\left(\sup_x (\min(\mu_{A'}(x), \mu_A(x))), \mu_B(y)\right).
 \end{aligned}
 \tag{1.28}$$

On en déduit la construction graphique du sous-ensemble flou B', présentée sur la figure suivante.

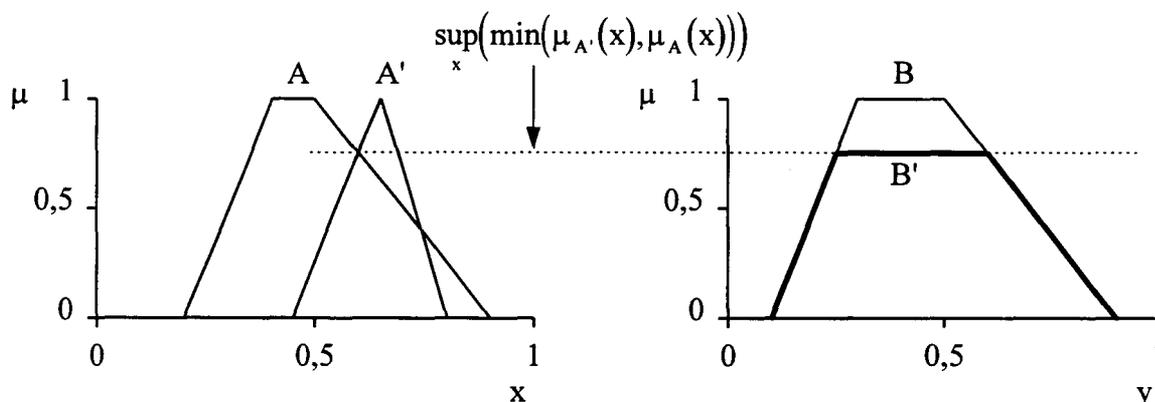


Figure 1.8 : Construction du sous-ensemble flou B'.

2.6 Agrégation de règles floues

La description d'un système par un expert nécessite l'emploi de plusieurs règles floues R^i . Elles sont regroupées au sein d'une base de règles.

$$\text{Règles } R^i : \text{SI } x \text{ est } A^i \text{ ALORS } y \text{ est } B^i, \tag{1.29}$$

où A^i et B^i , $i = 1$ à m , sont des sous-ensembles flous parfaitement définis sur leurs univers de discours respectifs U et V .

Le problème posé est identique au précédent. En effet, à partir d'une information provenant du monde extérieur (i.e. un sous-ensemble flou A') et de cette base de règles, on désire déterminer une décision B' .

$$\begin{aligned} & \text{Fait flou : } x \text{ est } A', \\ & m \text{ Règles } R^i : \text{SI } x \text{ est } A^i \text{ ALORS } y \text{ est } B^i, \\ & \text{Conclusion : } y \text{ est } B'. \end{aligned} \quad (1.30)$$

Le plus souvent, la description sous forme de règles d'un système physique est disjonctive. C'est-à-dire que le système est décrit comme une union des cas envisagés dans les règles. Par exemple, pour la régulation de la température d'un bain, la base de règle suivante peut être envisagée :

$$\begin{aligned} & R^1 : \text{SI la température de l'eau est } \textit{élevée} \text{ ALORS ajouter de l'eau } \textit{froide}, \\ & \text{OU } R^2 : \text{SI la température de l'eau est } \textit{moyenne} \text{ ALORS ne rien faire,} \\ & \text{OU } R^3 : \text{SI la température de l'eau est } \textit{faible} \text{ ALORS ajouter de l'eau } \textit{chaude}. \end{aligned} \quad (1.31)$$

Si la description du système est disjonctive, la relation globale R le décrivant est définie en utilisant un opérateur d'union sur les règles floues R^i .

$$R = R^1 \vee \dots \vee R^m. \quad (1.32)$$

Ainsi, cette relation globale R se caractérise par la fonction d'appartenance μ_R :

$$\begin{aligned} \mu_R : U \times V & \rightarrow [0 ; 1] \\ (x, y) & \mapsto \mu_R(x, y) = \bigcup_{i=1}^m \mu_{R^i}(x, y) = \mu_{R^1}(x, y) \vee \dots \vee \mu_{R^m}(x, y). \end{aligned} \quad (1.33)$$

A partir de cette relation globale R , on peut appliquer le mécanisme d'inférence défini dans la section 2.5 de ce chapitre. La fonction d'appartenance de la décision B' est alors :

$$\begin{aligned} \mu_{B'}(y) & = \sup_x (\mu_{A'}(x) \wedge \mu_R(x, y)) \\ & = \sup_x (\mu_{A'}(x) \wedge (\mu_{R^1}(x, y) \vee \dots \vee \mu_{R^m}(x, y))), \end{aligned} \quad (1.34)$$

où \wedge est une T-norme caractérisant l'opérateur du produit cartésien, \vee est une T-conorme définissant l'opérateur d'union et la fonction \sup est l'opérateur de la projection floue.

Remarque :

Si le sous-ensemble flou A' correspond à l'un des sous-ensembles flous utilisé dans la base de règles, par exemple $A' = A^i$, la conclusion B' peut être différente de B^i . En effet, afin d'obtenir une partition convenable de l'univers de discours U , l'intersection entre deux sous-ensembles flous successifs A^i et A^{i+1} est non nulle (figure 1.9).

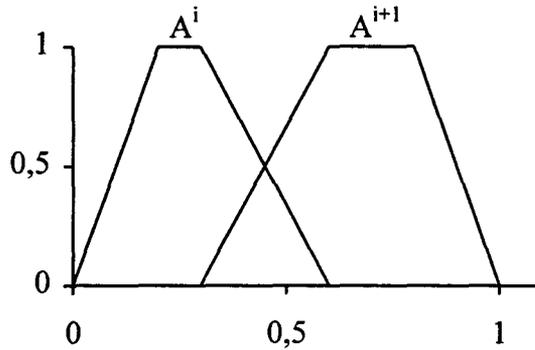


Figure 1.9 : Deux sous-ensembles flous successif A^i et A^{i+1} .

Par conséquent, la règle R^i , mais aussi les règles R^{i-1} et R^{i+1} sont activées. De ce fait, la conclusion B' , qui est l'union des solutions élémentaires B^j , $j = 1$ à m , est différente de B^i . La figure qui suit présente la conclusion B^{i+1} générée par la règle R^{i+1} (SI x est A^{i+1} ALORS y est B^{i+1}).

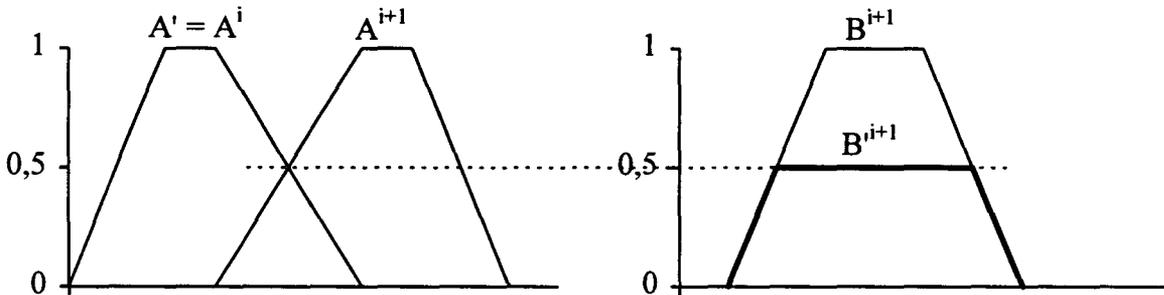


Figure 1.10 : Conclusion élémentaire pour la règle R^{i+1} .

3 Commande floue

3.1 Introduction

Dans un premier temps, la théorie des sous-ensembles flous a été développée pour des besoins de modélisation de systèmes complexes. Ceux-ci étaient décrits sous la forme d'une base de règles, transcription de l'analyse d'un expert. Pourtant, la réalisation d'une commande par l'inversion de cette base de règles pose des problèmes d'incertitudes et l'on aboutit souvent à des décisions arbitraires.

Par la suite, les techniques utilisées pour la modélisation ont été appliquées afin de transcrire directement la stratégie de commande de l'expert.

3.2 Commande floue

3.2.1 Structure d'une commande floue

Lee a présenté dans [Lee 1990] la structure conventionnelle d'une commande floue (figure 1.11). Celle-ci est composée de quatre blocs distincts dont les définitions sont données plus bas.

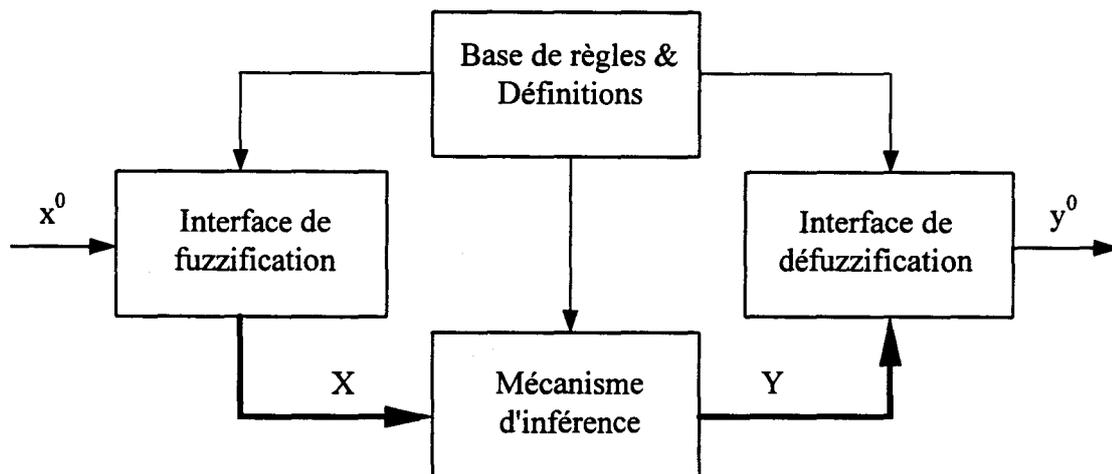


Figure 1.11 : Structure d'une commande floue.

3.2.2 Base de règles et définitions

Bien que n'ayant pas d'existence réelle, on regroupe dans ce bloc l'ensemble les définitions utilisées dans la commande floue (univers de discours, partition floue, choix des

opérateurs, ...), ainsi que la base de règles, transcription sous forme de règles floues de la stratégie de commande fournie par l'expert.

3.2.2.1 Partition floue

La création et l'utilisation d'une base de règles nécessite l'existence, pour chaque univers de discours considéré, de sous-ensembles flous particuliers. La partition floue consiste à définir ces sous-ensembles flous de façon à recouvrir l'univers de discours.

Dans la théorie des ensembles, une partition consiste à définir, sur un ensemble U , n sous-ensembles F_i ayant les propriétés suivantes :

$$\begin{aligned} \bigcup_{i=1}^n F_i &= U \\ \forall i \neq j; F_i \cap F_j &= \emptyset. \end{aligned} \quad (1.35)$$

En logique floue, l'union de sous-ensembles flous est un sous-ensemble flou caractérisé par une fonction d'appartenance à valeurs dans $[0 ; 1]$. Par conséquent, la partition floue par n sous-ensembles flous F_i d'un univers de discours U ne peut être considérée qu'à un niveau ε . Ce niveau ε représente le degré minimum d'appartenance d'un élément x de l'univers de discours U à l'union des sous-ensembles flous F_i . Effectuer une partition floue de niveau ε consiste donc à définir n sous-ensemble flous satisfaisant :

$$\begin{aligned} \bigcup_{i=1}^n F_i &\supseteq U_\varepsilon \\ \text{avec } U_\varepsilon &= \{x \in U; \mu_{U_\varepsilon}(x) = \varepsilon\}. \end{aligned} \quad (1.36)$$

Cette condition se traduit au niveau des fonctions d'appartenance par :

$$\forall x \in U; \mu_{F_1}(x) \vee \dots \vee \mu_{F_n}(x) \geq \varepsilon, \quad (1.37)$$

où \vee est une T-conorme utilisée comme opérateur d'union.

Comme la fonction *max* minore toutes les T-conormes (Remarque 1 sur les opérateurs d'union), pour assurer une partition floue de niveau ε , il faut et il suffit que :

$$\min_{x \in U} \max_{i=1 \text{ à } n} (\mu_{F_i}(x)) \geq \varepsilon. \quad (1.38)$$

Par exemple, si l'on considère l'univers de discours d'une température comme étant l'intervalle $[0^{\circ}\text{C} ; 60^{\circ}\text{C}]$, on peut choisir trois sous-ensembles flous représentatifs de la température ambiante *Faible*, *Moyenne* et *Elevée* de cet univers de discours présentés sur la figure suivante :

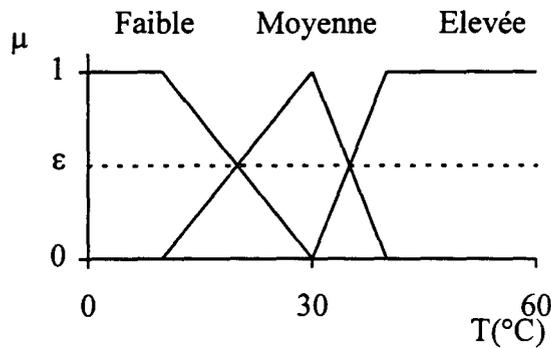


Figure 1.12 : Exemple de partition floue.

3.2.2.2 Base de règles

Une fois la partition des différents univers de discours réalisée, il est possible de définir la base de règles. Celle-ci définit les relations entre les classes d'événements considérés en entrée et les commandes correspondantes.

Si l'on considère n univers de discours U_i pour les prémisses des règles floues et si pour chaque univers U_i on définit une partition en m_i sous-ensembles flous, le nombre maximum de règles r_{\max} est :

$$r_{\max} = \prod_{i=1}^n m_i . \quad (1.39)$$

Le nombre de règles définies par l'expert peut être inférieur à r_{\max} . C'est le cas, en particulier, s'il existe des configurations des prémisses des règles floues non pertinentes. Par exemple dans le cas d'un freinage automatique de véhicule la configuration *Vitesse importante ET Distance de l'obstacle nulle* n'est pas considérée.

De plus, le nombre de sous-ensembles flous définis sur l'univers de discours de la commande n'est pas forcément égal au nombre de règles. En effet, il est possible de considérer des combinaisons de sous-ensembles flous différentes aboutissant à la même conclusion. Par exemple, dans le cas de la commande automatique de freinage, les cas *Vitesse petite ET*

Distance de l'obstacle faible et Vitesse moyenne ET Distance de l'obstacle importante amènent toutes deux à la conclusion *Freinage faible*.

Enfin, on peut remarquer qu'une augmentation de la sensibilité de la commande floue obtenue par une partition plus fine (en augmentant le nombre m_i de sous-ensembles flous) des univers de discours des prémisses aboutit à un accroissement important du nombre de règles à définir.

3.2.3 Interface de fuzzification

Tous les opérateurs utilisés dans la commande floue agissent sur des sous-ensembles flous. Pourtant, les mesures effectuées sont des valeurs non floues. Par conséquent, il est nécessaire de transformer ces variables en des sous-ensembles flous d'un univers de discours correspondant à leurs domaines d'évolution. Pour ce faire, on utilise un opérateur dit de fuzzification qui associe à une mesure de la variable x^0 une fonction d'appartenance particulière $\mu_{x^0}(x)$.

Définition 1.11 :

L'opérateur de fuzzification F est une transformation de U dans U qui à une valeur x^0 associe le sous-ensemble flou X^0 caractérisé par sa fonction d'appartenance $\mu_{x^0}(x)$.

$$\begin{aligned}
 F: U &\rightarrow U \\
 x^0 &\mapsto F(x^0) = X^0 = \{x \in U; \mu_{x^0}(x)\}.
 \end{aligned}
 \tag{1.40}$$

Le choix de l'opérateur de fuzzification dépend fortement de la confiance que l'on accorde aux mesures effectuées. Ainsi, si la mesure de la variable x^0 est exacte, le sous-ensemble flou X^0 doit être représenté par un fait précis. Par conséquent, on utilise comme opérateur de fuzzification la transformation dite du singleton. La fonction d'appartenance $\mu_{x^0}(x)$ du sous-ensemble flou X^0 est définie par :

$$\begin{aligned}
 \mu_{x^0}: U &\rightarrow U \\
 x &\mapsto \mu_{x^0}(x) = 1 \text{ si } x = x^0 \\
 &\mu_{x^0}(x) = 0 \text{ si } x \neq x^0.
 \end{aligned}
 \tag{1.41}$$

La figure suivante montre l'aspect de cette fonction d'appartenance.

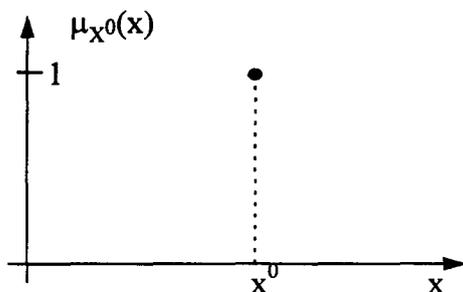


Figure 1.13 : Méthode de fuzzification pour une mesure exacte.

Si la mesure de la variable est imprécise, par exemple à cause de bruits, le sous-ensemble flou X^0 doit être représenté par un fait imprécis. On utilise donc une méthode de fuzzification qui associe à la variable x^0 une fonction d'appartenance telle que, par exemple :

$$\mu_{x^0}: U \rightarrow U$$

$$x \mapsto \mu_{x^0}(x) = \max\left(0, 1 - \frac{|x - x^0|}{\varepsilon}\right). \tag{1.42}$$

La représentation graphique de cette fonction d'appartenance est :

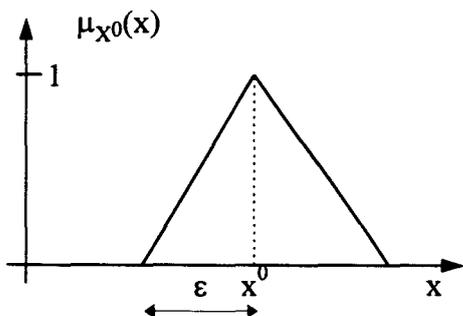


Figure 1.14 : Méthode de fuzzification pour une mesure imprécise.

La base du triangle ε est fonction de l'importance relative des erreurs commises sur la mesure de x^0 . En effet, plus elles sont importantes, plus la mesure de la variable x^0 devient imprécise, et donc, plus le triangle doit s'élargir. Pour les cas extrêmes, si il n'y a pas de bruits ($\varepsilon = 0$) on retrouve la fuzzification par la méthode du singleton ; si on ne peut accorder aucune confiance à la mesure effectuée ($\varepsilon = \infty$) le sous-ensemble flou X^0 devient complètement incertain sur l'univers de discours considéré (et alors $X^0 = U$).

3.2.3 Mécanisme d'inférence

A partir de la base de règles (fournie par l'expert) et du sous-ensemble flou X^0 correspondant à la fuzzification de la variable mesurée $x^0 = [x^0_1, \dots, x^0_n]^T \in U \subseteq \mathfrak{R}^n$, le mécanisme d'inférence calcule le sous-ensemble flou Y relatif à la commande à appliquer au système.

$$\begin{array}{l} x \text{ est } X^0, \\ m \text{ règles } R^i : \text{SI } x_1 \text{ est } X^i_1 \text{ ET ...ET } x_n \text{ est } X^i_n \text{ ALORS } y \text{ est } Y^i, \\ y \text{ est } Y, \end{array} \quad (1.43)$$

où X^i_j , $i = 1$ à m et $j = 1$ à n , est le sous-ensemble flou de la règle R^i sur l'univers de discours de la $j^{\text{ème}}$ composante du vecteur x ; Y^i est le sous-ensemble flou relatif à la commande à appliquer pour la règle R^i . On utilise l'opérateur d'inférence sur l'union des m règles floues R^i et on obtient :

$$\begin{aligned} \mu_Y(y) &= \sup_x (\mu_{X^0}(x) \wedge_1 \mu_R(x, y)) \\ &= \sup_x (\mu_{X^0}(x) \wedge_1 (\mu_{R^1}(x, y) \vee \dots \vee \mu_{R^m}(x, y))), \end{aligned} \quad (1.44)$$

où :

$$\mu_{R^i}(x, y) = (\mu_{X^i_1}(x_1) \wedge_2 \dots \wedge_2 \mu_{X^i_n}(x_n)) \wedge_3 \mu_{Y^i}(y). \quad (1.45)$$

Dans ces deux équations, les opérateurs \wedge_i , $i = 1$ à 3 , sont T-normes (voir la section sur les opérateurs d'intersection). L'opérateur \wedge_1 correspond au produit cartésien ; \wedge_2 représente la fonction *ET* utilisée dans les prémisses des règles floues ; \wedge_3 est une implication floue (décrite dans la section sur l'inférence floue).

3.2.4 Interface de défuzzification

Le sous-ensemble flou Y de l'univers de discours V ayant été calculé par le mécanisme d'inférence, l'interface de défuzzification a pour objectif de le transformer en une valeur non floue permettant ainsi la commande effective du système. Cette opération est appelée défuzzification.

Définition 1.12 :

L'opérateur de défuzzification D_F est une fonction d'un univers de discours V vers lui-même qui à un sous-ensemble flou Y , caractérisé par sa fonction d'appartenance μ_Y , fait correspondre la valeur non floue y^0 :

$$D_F : V \rightarrow V$$

$$Y = \{y \in V; \mu_Y(y)\} \mapsto D_F(Y) = y^0. \quad (1.46)$$

Afin de présenter les méthodes de défuzzification les plus utilisées, on considère le sous-ensemble flou Y de l'univers de discours V suivant :

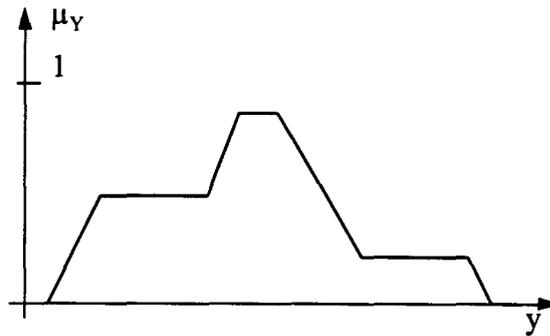


Figure 1.15 : Définition du sous-ensemble flou Y .

Cette fonction d'appartenance μ_Y représente la pertinence de chaque action y envisagée (i.e. appartenant à l'univers de discours V considéré).

La première méthode est l'utilisation du principe du maximum. Celui-ci consiste à choisir comme sortie y^0 de l'interface de défuzzification une des valeurs y possédant la plus grande appartenance au sous-ensemble flou Y . Cette méthode revient à choisir l'une des actions les plus pertinentes proposées par le mécanisme d'inférence.

$$D_F : V \rightarrow V$$

$$Y = \{y \in V; \mu_Y(y)\} \mapsto D_F(Y) = y^0 \text{ avec } \mu_Y(y^0) = \sup_y(\mu_Y(y)). \quad (1.47)$$

Ainsi, pour le sous-ensemble flou Y de la figure 1.15, la figure suivante donne, pour cette méthode, la construction de la sortie y^0 .

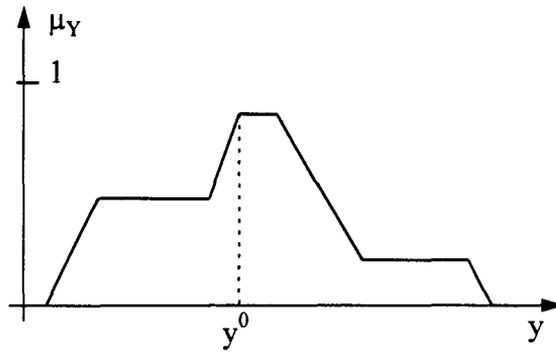


Figure 1.16 : Sortie de la commande floue pour une défuzzification par le principe du maximum.

Il se peut, comme dans le cas de la figure 1.15, que la fonction d'appartenance μ_Y du sous-ensemble flou Y possède plusieurs maxima. Dans ce cas, et afin d'éviter un choix arbitraire de la valeur y^0 , on choisit comme opérateur de défuzzification une transformation qui effectue la moyenne de ces maxima, c'est-à-dire une moyenne des actions les plus pertinentes.

$$D_F : V \rightarrow V$$

$$Y = \{y \in V; \mu_Y(y)\} \mapsto D_F(Y) = y^0 = \frac{\int y dy}{\int dy} \quad (1.48)$$

$$\text{avec } S = \left\{ y_i \in V; \mu_Y(y_i) = \sup_y (\mu_Y(y)) \right\}.$$

La figure suivante présente, pour cette méthode, la construction de la sortie y^0 pour le sous-ensemble flou Y de la figure 1.15.

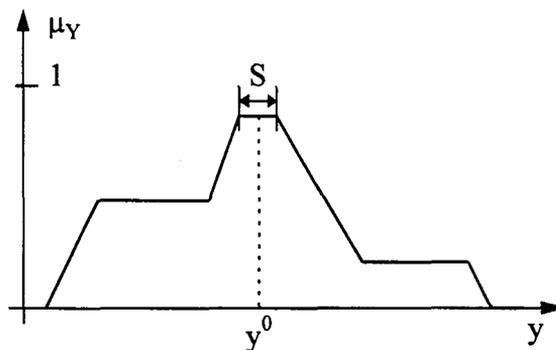


Figure 1.17 : Sortie de la commande floue pour une défuzzification par le principe de la moyenne des maxima.

Pour prendre en compte l'influence de l'ensemble des valeurs y proposées par la solution floue Y (i.e. possédant une appartenance non nulle à Y , $\mu_Y(y) \neq 0$), on utilise une méthode de défuzzification dite barycentrique.

$$D_F : V \rightarrow V$$

$$Y = \{y \in V; \mu_Y(y)\} \mapsto D_F(Y) = y^0 = \frac{\int_V y \mu_Y(y) dy}{\int_V \mu_Y(y) dy}. \quad (1.49)$$

On présente sur la prochaine figure l'application de cette méthode au sous-ensemble flou Y de la figure 1.15.

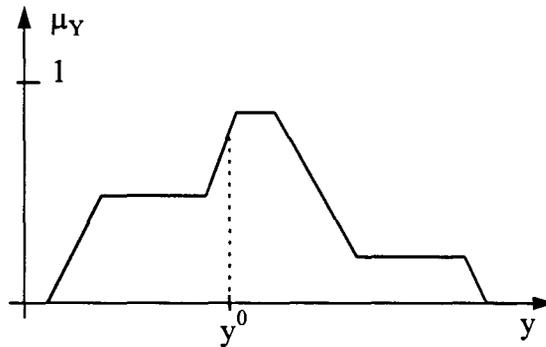


Figure 1.18 : Sortie de la commande floue pour une défuzzification barycentrique.

Remarque :

Dans le cas où les sous-ensembles flous Y^i relatifs aux commandes à appliquer dans les règles floues R^i sont des singletons (équation (1.50)), la méthode de défuzzification barycentrique se ramène à la transformation (1.51).

$$\forall i = 1, \dots, r; \mu_{Y^i}(y) = 1 \text{ si } y = y^i \\ = 0 \text{ si } y \neq y^i \quad (1.50)$$

$$D_F : V \rightarrow V$$

$$Y = \{y \in V; \mu_Y(y)\} \mapsto D_F(Y) = y^0 = \frac{\sum_{i=1}^r y^i \mu_Y(y^i)}{\sum_{i=1}^r \mu_Y(y^i)}, \quad (1.51)$$

où r est le nombre de sous-ensembles flous Y^i distincts sur l'univers de discours V .

3.3 Exemples d'application

3.3.1 Commande floue d'un système du premier ordre

3.3.1.1 Introduction

Le but de cet exemple très simple est de présenter les différentes étapes de la conception d'une commande floue, ainsi que la construction graphique de la commande pour une valeur particulière du vecteur de mesure.

3.3.1.2 Présentation du système

Le processus que l'on désire commander est un système du premier ordre dont la constante de temps τ et le gain statique k sont incertains, mais considérés positifs. Pour représenter cette contrainte, les paramètres de la commande floue seront établis pour des valeurs nominales de k et τ . La figure 1.19 présente le système étudié.

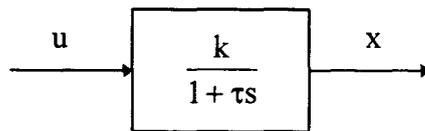


Figure 1.19 : Système du premier ordre.

où u est la commande du système et $x \in \mathcal{R}$ est l'état du système. Ils sont reliés par l'équation différentielle suivante :

$$\dot{x} = -\frac{1}{\tau}x + \frac{k}{\tau}u. \quad (1.52)$$

3.3.1.3 Analyse de l'expert

On désire commander le système par une boucle de régulation classique présentée sur la figure suivante :

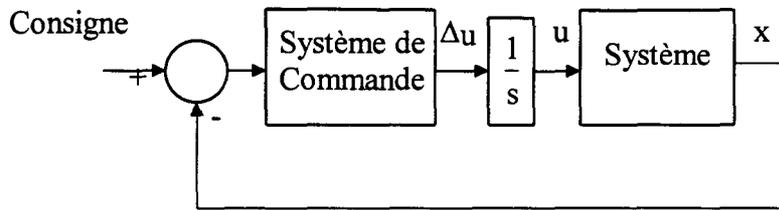


Figure 1.20 : Boucle de régulation utilisée.

Choix des variables :

Les variables de régulation sont l'erreur entre la consigne et la sortie du système et la variation de l'erreur.

$$\begin{aligned} \text{Erreur} = e &= \text{Consigne} - x \\ \text{Variation de l'erreur} = \Delta e &= e(t) - e(t - T) \end{aligned} \tag{1.53}$$

Partition des univers de discours :

On considère pour chaque variable de mesure deux sous-ensembles flous, *Positive* et *Négative*. Ils sont définis sur l'ensemble des réels. De plus, on admet trois actions possibles pour la variation de la sortie de la commande floue définies par les sous-ensembles flous *Négative*, *Nulle* et *Positive* sur l'univers de discours $[-\Delta U ; \Delta U]$. La figure 1.21 présente l'aspect de ces différents sous-ensembles flous.

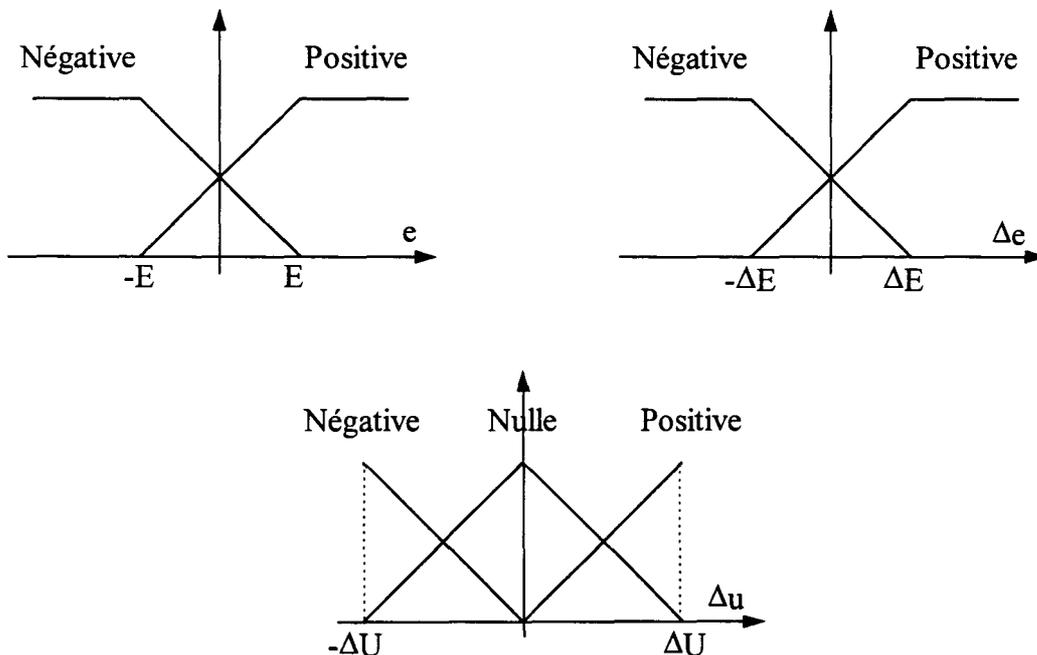


Figure 1.21 : Présentation des sous-ensembles flous utilisés.

Les différentes valeurs E, ΔE et ΔU sont des paramètres de réglage de la commande floue.

Base de règles :

Le système réagit de la façon à suivre les variations de la commande. Par conséquent, pour obtenir une erreur et une variation de l'erreur nulles, on utilise la base de règles définie dans le tableau 1.8.

Règle N° :	SI e est	ET Δe est	ALORS Δu est
1	Négative	Négative	Négative
2	Négative	Positive	Nulle
3	Positive	Négative	Nulle
4	Positive	Positive	Positive

Tableau 1.8 : Base de règles de la commande floue.

Choix des différents opérateurs :

La dernière étape de la conception d'une commande floue consiste à effectuer le choix des différents opérateurs. Ceux-ci sont regroupés dans le tableau 1.9.

Opérateur	Spécification	Choix
Intersection	T-norme	min
Produit Cartésien	T-norme	min
Implication floue	T-norme	min
Projection floue		sup
Union	T-conorme	max
Fuzzification		Singleton
Défuzzification		Barycentrique

Tableau 1.9 : Choix des différents opérateurs de la commande floue.

3.3.1.4 Calcul de la variation de la commande

La sortie Δu de la commande floue est obtenue par défuzzification du sous-ensemble flou caractérisé par la fonction d'appartenance :

$$\begin{aligned}\mu(x^0, \Delta u) &= \sup_x \min(\mu_{x^0}(x), \mu_R(x, \Delta u)) \\ &= \sup_x \min\left(\mu_{x^0}(x), \max_{i=1 \text{ à } 4}(\mu_{R^i}(x, \Delta u))\right).\end{aligned}\quad (1.54)$$

Dans cette équation, x est le vecteur de mesure $[e ; \Delta e]^T$ et X^0 représente le fait *e est E⁰ ET Δe est ΔE⁰*, où E^0 et ΔE^0 sont les sous-ensembles flous obtenus par fuzzification des variables mesurées e^0 et Δe^0 . La fuzzification choisie étant la méthode du singleton, on a :

$$\mu(x^0, \Delta u) = \max_{i=1 \text{ à } 4}(\mu_{R^i}(x^0, \Delta u)), \quad (1.55)$$

où la fonction d'appartenance du sous-ensemble flou défini par la règle R^i est :

$$\mu_{R^i}(x, \Delta u) = \min(\mu_{X^i}(x), \mu_{Y^i}(\Delta u)). \quad (1.56)$$

Ici, X^i représente le fait *e est Eⁱ ET Δe est ΔEⁱ*, où E^i et ΔE^i sont les sous-ensembles flous de la règle R^i . De même, Y^i est le sous-ensemble flou de la variation de la commande Δu pour la règle R^i .

$$\mu_{X^i}(x) = \min(\mu_{E^i}(e), \mu_{\Delta E^i}(\Delta e)). \quad (1.57)$$

En définitive, la fonction d'appartenance du sous-ensemble flou représentant la variation de la commande à appliquer au système est :

$$\mu(e^0, \Delta e^0, \Delta u) = \max_{i=1 \text{ à } 4}(\min(\mu_{E^i}(e^0), \mu_{\Delta E^i}(\Delta e^0), \mu_{Y^i}(\Delta u))). \quad (1.58)$$

En appliquant la défuzzification barycentrique, la variation de la commande Δu devient :

$$\Delta u(e^0, \Delta e^0) = \frac{\int_{-\Delta U}^{\Delta U} \Delta u \mu(e^0, \Delta e^0, \Delta u) d\Delta u}{\int_{-\Delta U}^{\Delta U} \mu(e^0, \Delta e^0, \Delta u) d\Delta u}. \quad (1.59)$$

3.3.1.4 Construction graphique de la sortie de la commande floue

En reprenant la formulation (1.55), on s'aperçoit que le sous-ensemble flou relatif à la variation de la commande Δu correspond au maximum des commandes élémentaires $\mu_{R^i}(x^0, \Delta u)$, définies à partir du vecteur de mesure x^0 . Ces commandes élémentaires sont construites en utilisant (1.56). Pour résumer, la construction graphique de la solution floue suit les étapes suivantes :

- 1- Pour chaque règle R^i , définir $\mu_{E^i}(e^0)$ et $\mu_{\Delta E^i}(\Delta e^0)$ (Fuzzification par singleton).
- 2- Reporter le minimum des deux valeurs sur le sous-ensemble flou Y^i (Opérateur ET).
- 3- Construire la commande floue élémentaire pour la règle R^i (Implication floue).
- 4- Construire le sous-ensemble flou solution en prenant le maximum des solutions élémentaires (Agrégation des règles par l'opérateur d'union ici la fonction *max*).
- 5- Appliquer l'opérateur de défuzzification sur ce sous-ensemble flou afin de déterminer la commande à appliquer.

Ces étapes de la construction graphique de la commande Δu^0 pour une mesure particulière e^0 et Δe^0 ont présentées figure 1.22.

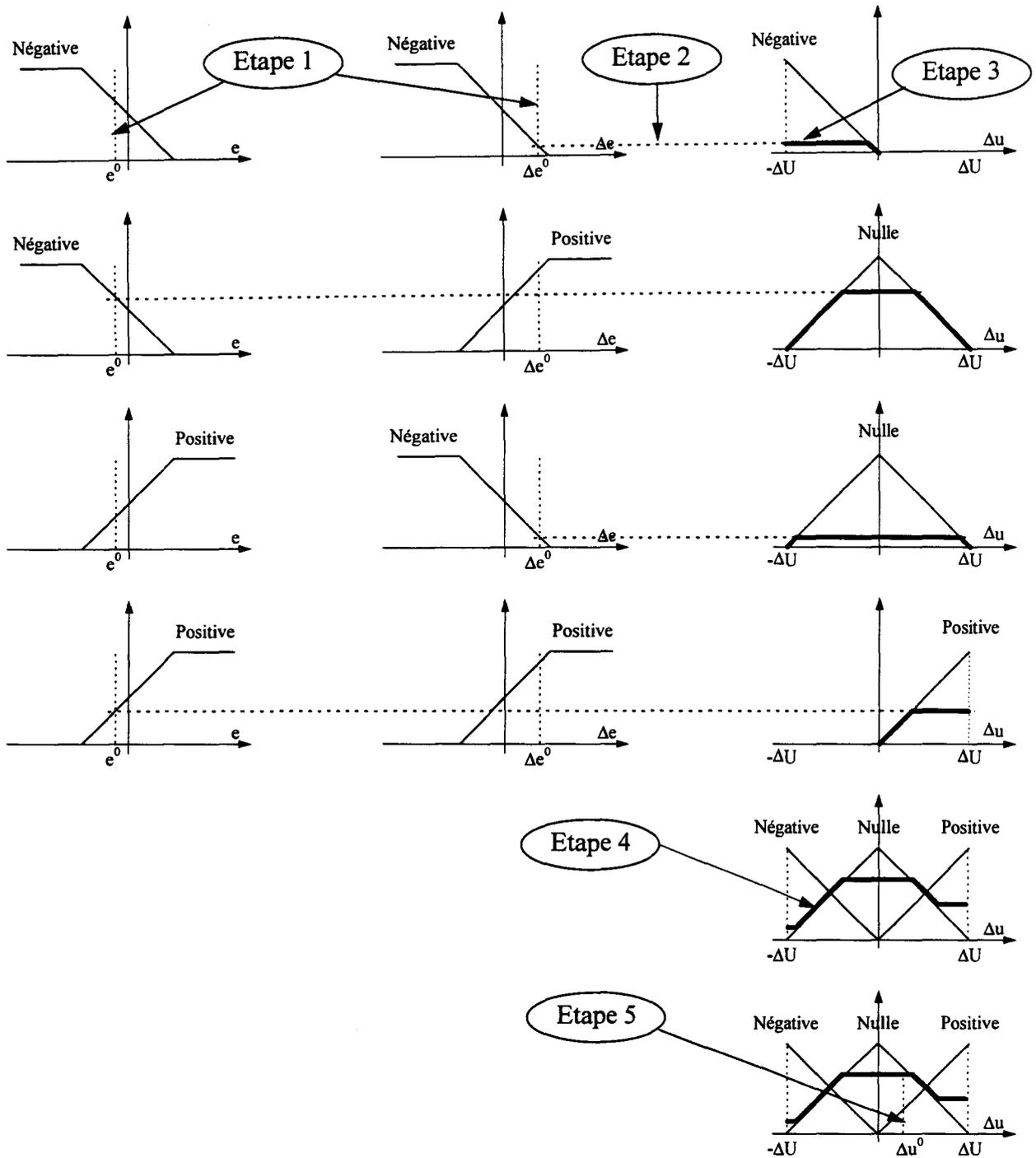


Figure 1.22 : Construction de la commande floue Δu_0 .

3.3.1.5 Simulations

La première étape consiste à déterminer les paramètres E , ΔE et U de la commande floue dans le cas d'un système du premier ordre dont la constante de temps τ et le gain statique k sont connus. Ces paramètres sont choisis afin d'obtenir les résultats souhaités. La figure qui suit présente les résultats de simulation pour le système nominal ($\tau = 1$ s et $k = 1$).

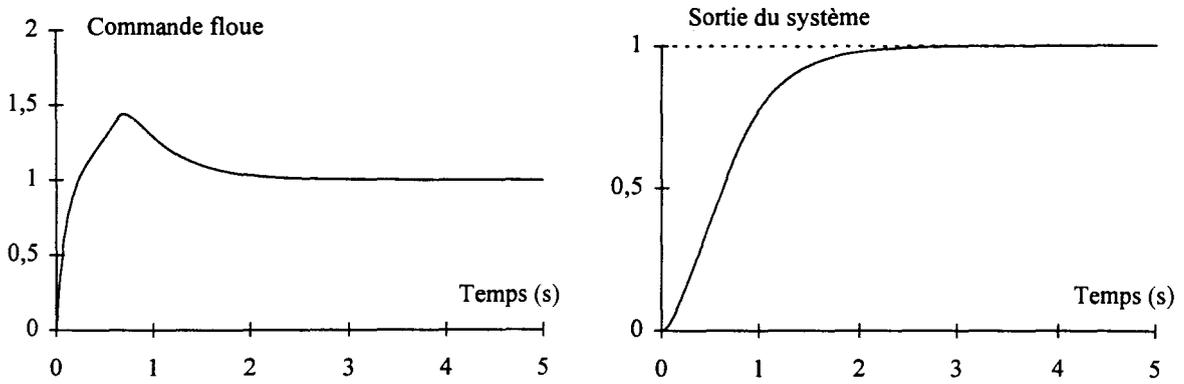


Figure 1.23 : Simulations pour le système nominal $\tau = 1s, k = 1$.

Une fois les paramètres fixés pour obtenir les résultats désirés et afin de tester la robustesse de la commande floue réalisée, on effectue, avec la même loi de commande, de nouvelles simulations pour lesquelles les valeurs du gain statique et la constante de temps du système sont différentes. Deux cas extrêmes ($k = 0,5$ et $\tau = 1,5s$; $k = 1,5$ et $\tau = 0,5$) sont présentés sur les deux figures suivantes.

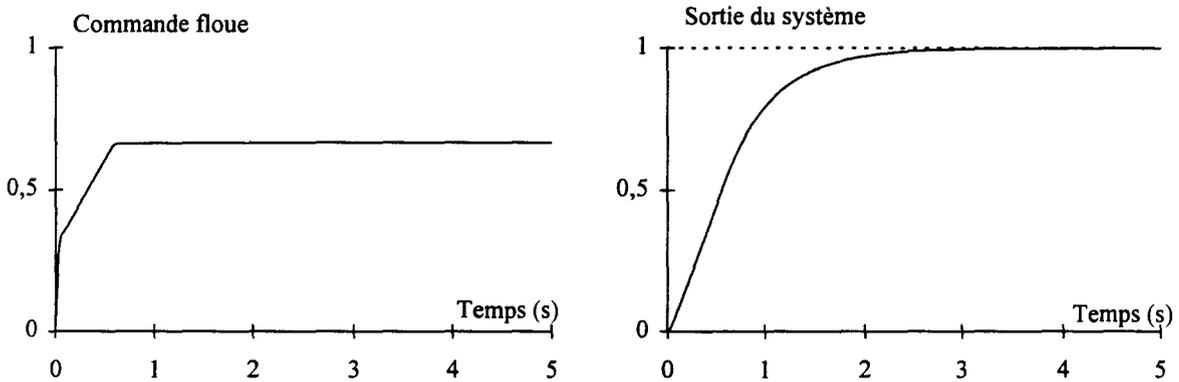


Figure 1.24 : Simulations pour $\tau = 0,5s$ et $k = 1,5$.

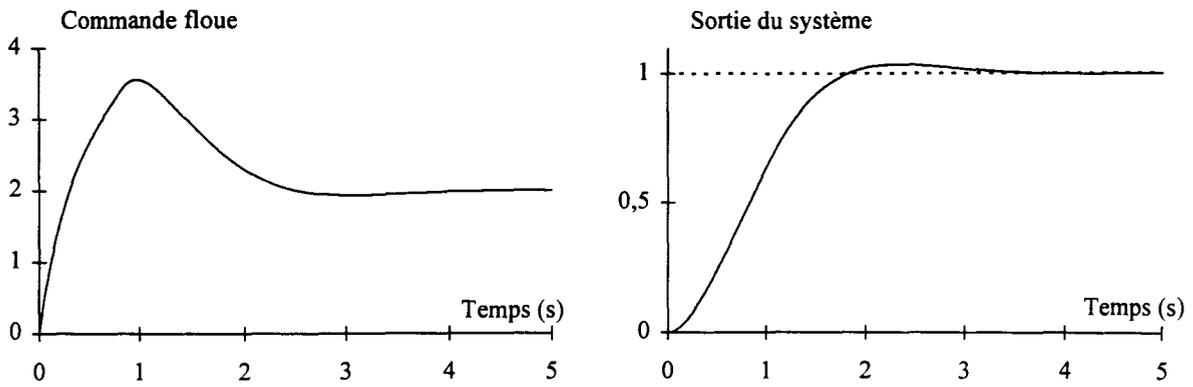


Figure 1.25 : Simulations pour $\tau = 1,5s$ et $k = 0,5$.

3.3.2 Commande floue du système du pendule inversé

3.3.2.1 Introduction

Le but de cet exemple est de montrer la simplicité de mise en oeuvre d'une commande floue pour un système non-linéaire.

3.3.2.2 Présentation du système

Le système que l'on désire commander est un pendule inversé présenté figure 1.26. L'objectif de la commande est de ramener la barre en position d'équilibre ($\theta = 0$).

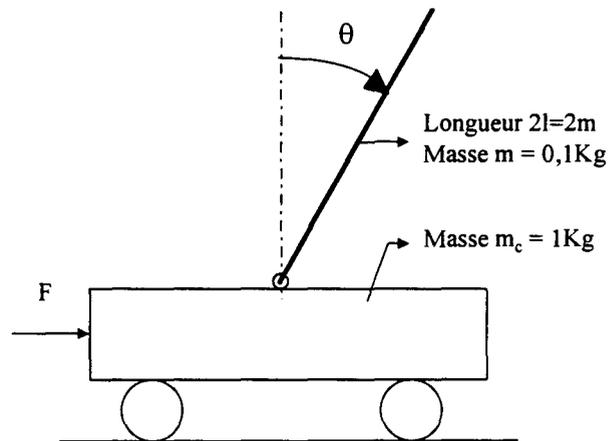


Figure 1.26 : Système du pendule inversé.

Sur cette figure, F est la force appliquée au chariot et θ est l'angle de la barre avec la verticale.

Le pendule inversé obéit à l'équation différentielle suivante :

$$\ddot{\theta} = \frac{g \sin \theta - \cos \theta \left(\frac{F + m l \dot{\theta}^2 \sin \theta}{m + m_c} \right)}{l \left(\frac{4}{3} - \frac{m \cos^2 \theta}{m + m_c} \right)}. \quad (1.60)$$

3.3.2.3 Analyse de l'expert

La boucle de régulation utilisée est la suivante :

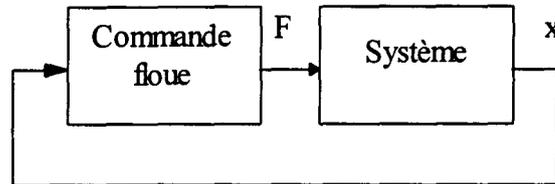


Figure 1.27 : Boucle de régulation.

Sur cette figure, x est le vecteur de mesure.

Choix des variables :

Les variables utilisées dans la commande sont l'angle θ et la vitesse angulaire $d\theta/dt$. Par conséquent, le vecteur de mesure est :

$$x = [\theta ; d\theta/dt]^T \quad (1.61)$$

Partition des univers de discours :

On définit deux sous-ensembles flous sur les univers de discours de θ (*Positif* et *Négatif*) et de $d\theta/dt$ (*Négative* et *Positive*). On considère trois actions de commande F : *Positive*, *Nulle* et *Négative* sur l'univers de discours $[-F_{\max} ; F_{\max}]$. La figure 1.28 présente la forme de ces sous-ensembles flous. Comme pour le premier exemple, les valeurs θ_{\max} , $d\theta/dt_{\max}$ et F_{\max} sont des paramètres de réglage essentiels de la commande floue.

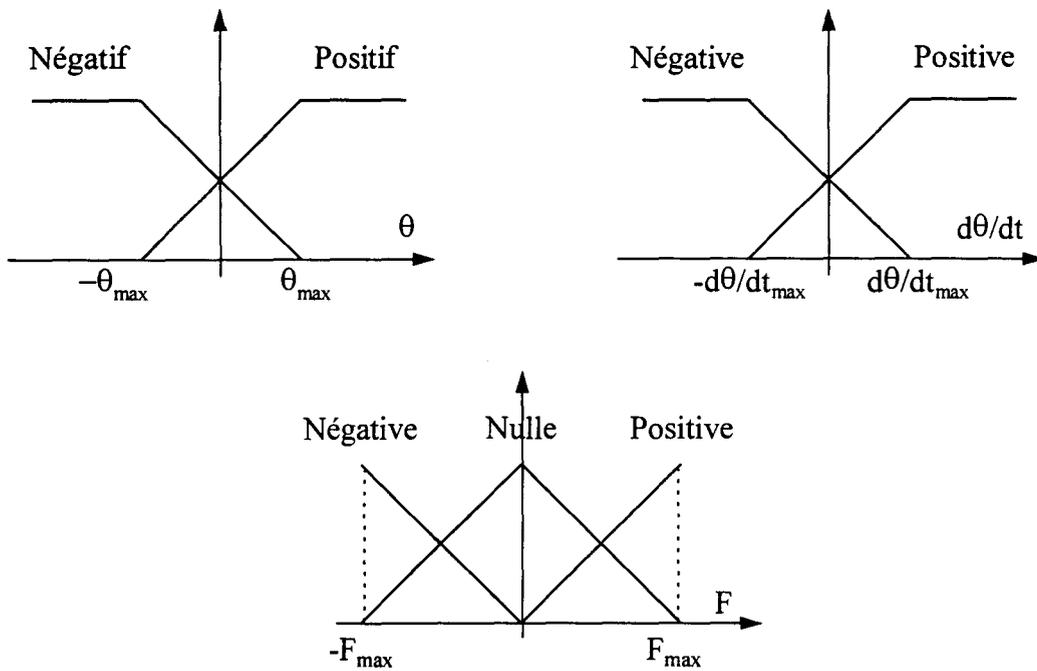


Figure 1.28 : Sous-ensembles flous utilisés pour la commande du pendule inversé.

Base de règles :

De manière pratique, on conçoit que plus on pousse le chariot (action F Positive) plus le pendule aura tendance à "revenir en arrière" (l'angle θ diminue) et inversement si l'on tire sur le chariot. Par conséquent, afin de ramener le pendule en position verticale, on définit la base de règles suivante :

Règle N° :	SI θ est	ET $d\theta/dt$ est	ALORS F est
1	Négatif	Négative	Négative
2	Négatif	Positive	Nulle
3	Positif	Négative	Nulle
4	Positif	Positive	Positive

Tableau 1.10 : Base de règles de la commande floue du pendule inversé.

Choix des opérateurs :

Les opérateurs utilisés pour cette commande sont identiques à ceux décrit dans le tableau 1.9.

3.3.2.4 Simulations

Les paramètres de la commande floue sont fixés pour le système nominal présenté sur la figure 1.26. La figure suivante présente, dans ce cas, l'évolution de l'angle θ et de la commande F .

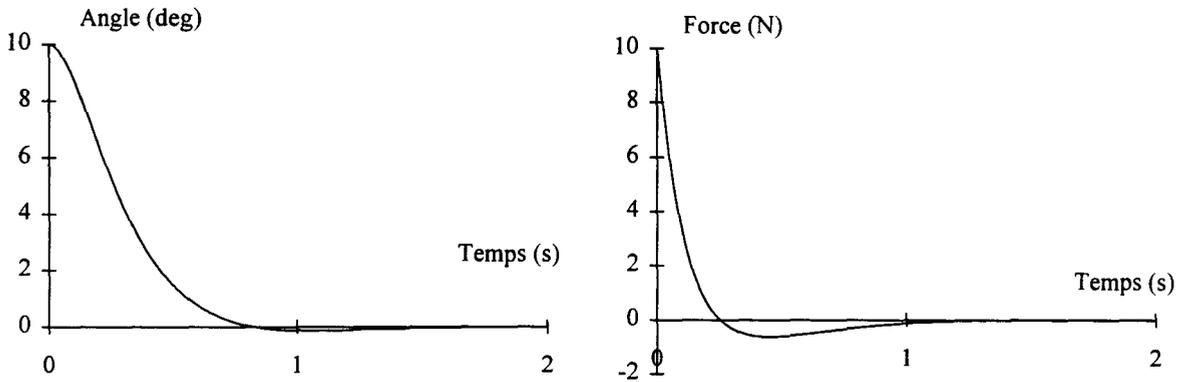


Figure 1.29 : Simulations pour le pendule inversé avec $[m ; m_c ; l] = [0.1 \text{ Kg} ; 1 \text{ Kg} ; 1 \text{ m}]$.

Afin de tester la robustesse de la commande ainsi réalisée, on applique la même loi de commande pour un système dont les paramètres essentiels (masse du chariot, masse et longueur de la barre) sont différents des paramètres nominaux.

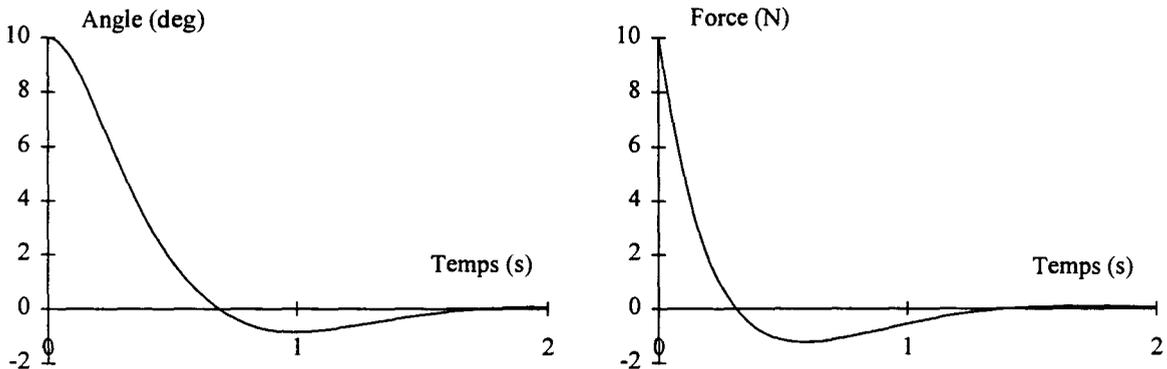


Figure 1.30 : Simulations pour le pendule inversé avec $[m ; m_c ; l] = [0.2 \text{ Kg} ; 0.5 \text{ Kg} ; 2 \text{ m}]$.

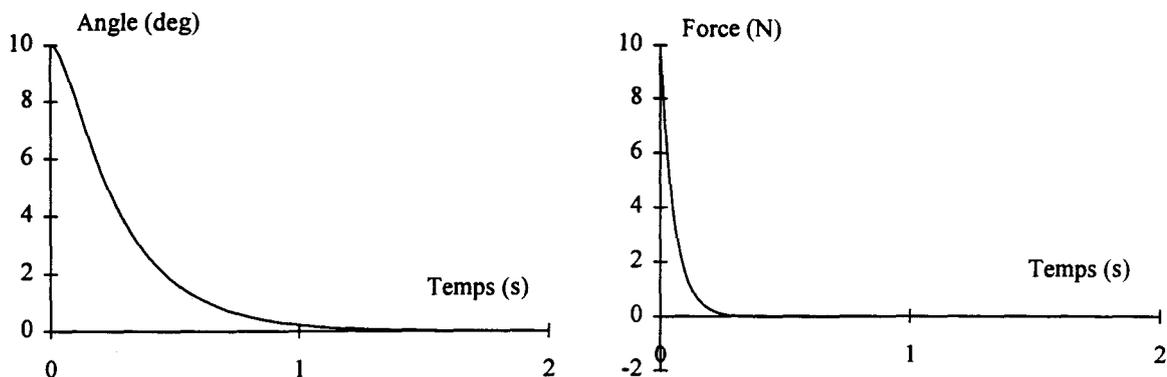


Figure 1.31 : Simulations pour le pendule inversé avec $[m ; m_c ; l] = [0.05 \text{ Kg} ; 2 \text{ Kg} ; 0.5 \text{ m}]$.

Les deux figures précédentes présentent les simulations effectuées pour deux cas extrêmes. Ces simulations montrent la capacité de la commande floue réalisée à absorber les erreurs de modélisation.

3.4 Conclusions

3.4.1 Avantages des commandes floues

Le principal avantage de la logique floue est la possibilité de prise en compte d'une information qualitative provenant de l'analyse d'un expert. La connaissance que possède un expert sur un système est en effet transcrite sous la forme d'une base de règles, le coeur même d'un système flou. L'ensemble des opérations utilisées pour obtenir la commande floue permet d'interpoler les commandes spécifiques aux règles définies par l'expert.

Un autre avantage des commandes floues est leur souplesse aussi bien lors de la conception qu'en cours d'utilisation. En effet, une commande floue ne nécessite pas la modélisation explicite du système. On peut donc réaliser une même architecture de commande floue commune à de nombreuses applications (la seule différence réside dans la base de règles). C'est pourquoi, on voit apparaître des circuits intégrés "flous" effectuant les opérations élémentaires (fuzzification, mécanisme d'inférence et défuzzification).

3.4.2 Inconvénients des commandes floues

Il existe trois principaux inconvénients à l'utilisation des commandes floues. Le premier découle de l'importance de la base de règles. En effet, il peut être difficile de transcrire sous forme de règles simples l'analyse d'un expert. En particulier, l'une des difficultés réside dans la

détermination de l'ensemble des variables (et par la suite les mesures de celles-ci) intervenant dans son jugement.

Le deuxième inconvénient concerne le réglage des différents paramètres. En effet, le choix de ces derniers est arbitraire et leurs influences sur les performances de la commande ne sont pas explicites. Par conséquent, dès que la base de règle s'étoffe, la durée de la phase de réglage devient généralement très importante.

Enfin, le dernier problème réside dans l'impossibilité de définir les limites de robustesse d'une commande floue. Par conséquent, et malgré les validations pratiques, les commandes floues sont souvent considérées comme "farfelues".

3.5 Objectifs fixés

Les objectifs des recherches entreprises furent de conserver les avantages des commandes floues tout en réduisant leurs inconvénients.

Pour ce faire, deux approches ont été envisagées. La première a consisté à considérer et à étudier la représentation non-linéaire d'un système flou. La deuxième, appelée commande floue multi-modèle, consiste à utiliser les propriétés d'interpolation de la logique floue pour effectuer des passages continus entre les commandes associées à différents modèles d'un même système.



Chapitre 2 : OPTIMISATION DES COMMANDES A BASE DE REGLES FLOUES

1 Introduction

Un système flou (figure 2.1) est, malgré son nom, un système parfaitement déterministe.

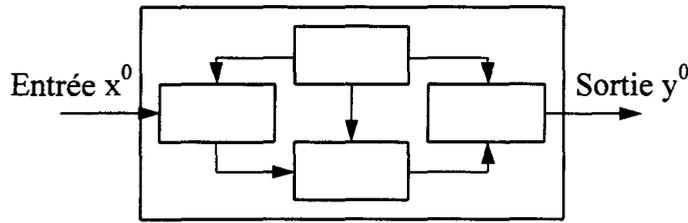


Figure 2.1 : Système flou.

En effet, pour un système flou défini à partir de m règles floues R^i de la forme :

$$\begin{aligned}
 & x \text{ est } X^0 \\
 m \text{ règles } R^i : & \text{SI } x_1 \text{ est } X_1^i \text{ ET } \dots \text{ ET } x_n \text{ est } X_n^i \text{ ALORS } y \text{ est } Y^i \\
 & y \text{ est } Y,
 \end{aligned} \tag{2.1}$$

où X_j^i et Y^i , $i = 1$ à m et $j = 1$ à n , sont respectivement pour la règle R^i le sous-ensemble flou de l'univers de discours de la $j^{\text{ème}}$ composante du vecteur état x et le sous-ensemble flou relatif à la sortie du système flou. Le sous-ensemble flou Y est caractérisé par la fonction d'appartenance :

$$\begin{aligned}
 \mu_Y(y) &= \sup_{x \in U} (\mu_{X^0}(x) \wedge_1 \mu_R(x, y)) \\
 &= \sup_{x \in U} (\mu_{X^0}(x) \wedge_1 (\mu_{R^1}(x, y) \vee \dots \vee \mu_{R^m}(x, y))),
 \end{aligned} \tag{2.2}$$

avec :

$$\mu_{R^i}(x, y) = (\mu_{X_1^i}(x_1) \wedge_2 \dots \wedge_2 \mu_{X_n^i}(x_n)) \wedge_3 \mu_{Y^i}(y). \tag{2.3}$$

Dans ces équations, les opérateurs \wedge_i sont des T-normes et \vee est une T-conorme. La valeur de sortie y est obtenue après l'application de l'opérateur de défuzzification D_F :

$$\begin{aligned}
 D_F : V &\rightarrow V \\
 Y = \{y \in V; \mu_Y(y)\} &\mapsto D_F(Y) = y^0.
 \end{aligned} \tag{2.4}$$

Par conséquent, un système flou peut être caractérisé par une fonction dépendant des choix effectués pour la définition des paramètres (P) (i.e. méthode de fuzzification, opérateurs

utilisés, méthode de défuzzification, définitions des sous-ensembles flous). La sortie y^0 est alors décrite par :

$$y^0 = f_p(x^0). \quad (2.5)$$

On peut donc envisager d'étudier un système flou par l'analyse de la fonction $f_p(x)$ le caractérisant. Cette approche a débuté par les travaux de Wang [Wan 1992a] et de Kosko [Kos 1992] qui ont mis en évidence les propriétés d'approximation de la logique floue.

Dans ce chapitre, ces travaux préliminaires seront présentés. Nous étendrons les propriétés qui en découlent à une approche logique qui passe par l'utilisation d'une T-conorme lors de l'agrégation des règles floues. De plus, l'analyse de l'influence de la méthode de fuzzification dans les performances d'un système flou sera exposée.

2 Historique

L'approche non linéaire de l'étude d'un système flou est très récente ([Wan 1992a], [Wan 1992b] [Buc 1992] [Zen 1994] [Kos 1992]). Ces différentes études ont montré qu'un système flou S peut approximer, dans un ensemble compact U et avec la précision désirée ε , toute fonction continue f de U .

$$\begin{aligned} \forall f \in U \text{ fonction continue d'un ensemble compact,} \\ \forall \varepsilon > 0, \exists S : \text{système flou,} \\ \forall x \in U, |f(x) - f_s(x)| \leq \varepsilon. \end{aligned} \tag{2.6}$$

2.1 Système flou considéré

Les études précédentes ont été réalisées pour un choix particulier de paramètres. Tout d'abord, la méthode de fuzzification utilisée est la méthode du singleton. Pour une mesure de l'entrée x^0 , la fonction d'appartenance μ_{x^0} du sous-ensemble flou X^0 est définie par :

$$\begin{aligned} \mu_{x^0} : U &\rightarrow U \\ x &\mapsto \mu_{x^0}(x) = 1 \text{ si } x = x^0 \\ &\mu_{x^0}(x) = 0 \text{ si } x \neq x^0. \end{aligned} \tag{2.7}$$

L'aspect de la fonction d'appartenance est :

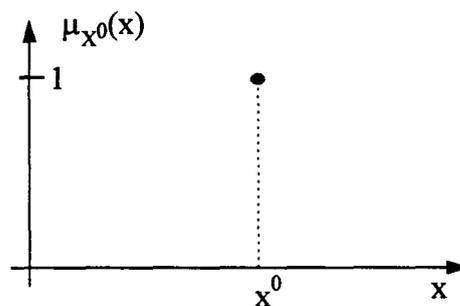


Figure 2.2 : Méthode du singleton.

Le choix de cette méthode implique que la mesure de x^0 est assimilée à un fait précis.

Ensuite, la T-norme utilisée pour tous les opérateurs d'intersection est la fonction produit. Ainsi, la fonction d'appartenance caractérisant la règle R^i est :

$$\mu_{R^i}(x, y) = \mu_{Y^i}(y) \prod_{j=1}^n \mu_{X_j^i}(x_j). \quad (2.8)$$

Enfin, les sous-ensembles flous Y^i de la sortie du système sont des singletons :

$$\begin{aligned} \forall i = 1, \dots, m; \mu_{Y^i}(y) &= 1 \text{ si } y = y^i \\ &= 0 \text{ si } y \neq y^i. \end{aligned} \quad (2.9)$$

Par conséquent, la méthode de défuzzification barycentrique utilisée donne pour sortie y^0 :

$$D_F: V \rightarrow V$$

$$Y = \{y \in V; \mu_Y(y)\} \mapsto D_F(Y) = y^0 = \frac{\sum_{i=1}^m y^i \mu_Y(y^i)}{\sum_{i=1}^m \mu_Y(y^i)}. \quad (2.10)$$

Ici, les règles floues sont considérées indépendantes les unes des autres et par conséquent, la sortie y^0 correspond à la moyenne pondérée de toutes les règles, même celles fournissant des conclusions identiques. Ce choix sous-entend une approche statistique du problème de la détermination du sous-ensemble flou de sortie Y (i.e. une seule source d'informations fournissant m conclusions). L'approche logique considère m sources d'informations (i.e. les m règles floues) donnant, au même moment, r conclusions distinctes. Dans ce cas, l'agrégation des règles doit être réalisée par l'union (i.e. l'application d'une T-conorme) de celles-ci.

Afin de montrer la différence entre ces deux approches, considérons une base de m règles floues avec $m-1$ règles donnant une conclusion C_1 presque fausse (i.e. la valeur d'appartenance de la sortie y à C_1 est faible) et une seule dont la conclusion est C_2 mais dont on est certain de la fiabilité (i.e. la valeur d'appartenance de la sortie y à C_2 est proche de 1). Pour un grand nombre de règles, l'utilisation d'une approche statistique tend à choisir la conclusion C_1 au détriment de C_2 . Si une T-conorme est utilisée pour agréger ces règles la conclusion finale sera comprise entre une valeur très proche de C_2 (cas où la T-conorme choisie est la fonction *max*) et $(C_1 + C_2)/2$. La figure qui suit présente, pour un cas particulier, l'évolution de la conclusion finale en fonction du nombre de règles floues utilisées.

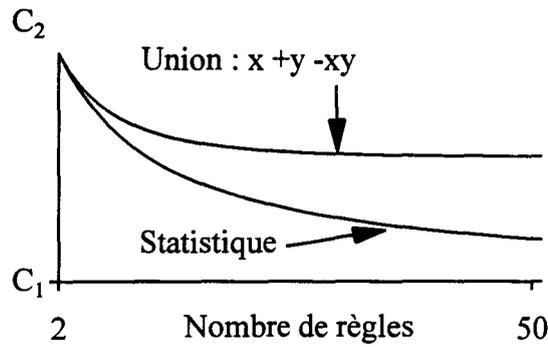


Figure 2.3 : Evolution de la conclusion finale en fonction du nombre de règles avec $\mu_1 = 0,1$ et $\mu_2 = 1$.

2.2 Détermination de la fonction f_S

Pour l'ensemble des choix réalisés dans la conception du système flou S, la sortie y^0 du système devient :

$$y^0 = \frac{\sum_{i=1}^m y^i \mu_Y(y^i)}{\sum_{i=1}^m \mu_Y(y^i)} = \sum_{i=1}^m y^i \frac{\mu_Y(y^i)}{\sum_{j=1}^m \mu_Y(y^j)}, \quad (2.11)$$

avec l'expression de $\mu_Y(y^i)$:

$$\mu_Y(y^i) = \mu_{Y^i}(y^i) \prod_{j=1}^n \mu_{X_j^i}(x_j) = \prod_{j=1}^n \mu_{X_j^i}(x_j). \quad (2.12)$$

Ainsi, le système flou S est défini par la fonction $f_S(x)$:

$$f_S(x) = \sum_{i=1}^m \xi^i(x) y^i, \quad (2.13)$$

où les fonctions $\xi^i(x)$ appelées FBF (Fuzzy Basis Function) sont données par :

$$\xi^i(x) = \frac{\prod_{j=1}^n \mu_{X_j^i}(x_j)}{\sum_{i=1}^m \prod_{j=1}^n \mu_{X_j^i}(x_j)} \quad (2.14)$$

et les $y^i \in \mathfrak{R}$, $i=1$ à m sont des constantes.

A titre d'illustration, considérons les sous-ensembles flous X^i suivants :

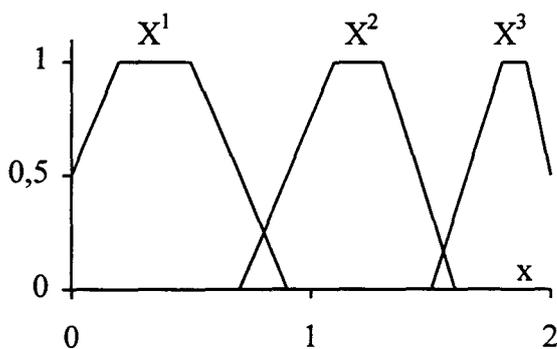


Figure 2.4 : Sous-ensembles flous considérés.

Pour ces sous-ensembles, on présente sur la figure suivante la forme des fonctions FBF correspondantes.

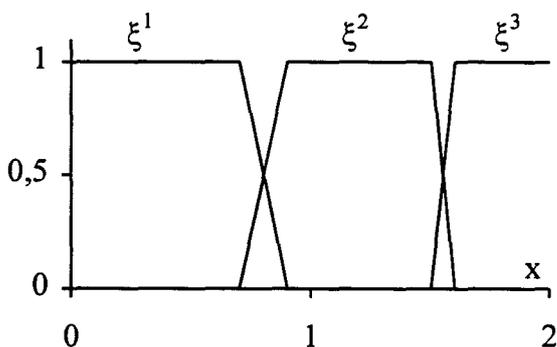


Figure 2.5 : Fonctions FBF ξ^i .

Théorème [Wan 1992a] :

Soit Z l'ensemble des fonctions de la forme (2.13), alors pour toute fonction f de l'ensemble compact $U \subset \mathbb{R}^n$ et pour toute constante arbitraire ε positif, il existe f_s de Z telle que :

$$\sup_{x \in U} |f(x) - f_s(x)| < \varepsilon \tag{2.15}$$

La démonstration de ce théorème, basée sur le théorème de Stone-Weierstrass, est présentée en annexe 1.

2.3 Réalisation d'une commande floue adaptative

2.3.1 Présentation

Wang a développé [Wan 1993] une commande floue adaptative stable d'un système non-linéaire en utilisant un système flou de la forme (2.13). Nous présentons ici la méthodologie que nous reprendrons par la suite.

Le système non-linéaire à commander est :

$$y^{(n)} = f(x) + bu, \quad (2.16)$$

où $x = [y, \dots, y^{(n-1)}]^T \in \mathfrak{R}^n$ est le vecteur d'état, f et b sont respectivement une fonction et une constante positive inconnues. Dans le cas d'un système parfaitement défini, avec $e = y_d - y$ l'erreur entre la sortie désirée et la sortie réelle du système, $E = [e, \dots, e^{(n-1)}]^T$, $k = [k_1, \dots, k_n]^T \in \mathfrak{R}^n$ les constantes telles que les racines du polynôme $h(s) = s^n + k_1s^{n-1} + \dots + k_{n-1}s + k_n$ sont à parties réelles négatives, la loi de commande

$$u^* = \frac{1}{b}(-f(x) + y_d^{(n)} + k^T E) \quad (2.17)$$

permet d'assurer la convergence de l'erreur vers 0. Comme le système est inconnu, on utilise un système flou à m règles de la forme :

$$R^i : \text{SI } x_1 \text{ est } X_1^i \text{ ET } \dots \text{ ET } x_n \text{ est } X_n^i \text{ ALORS } \theta \text{ est } \Theta^i. \quad (2.18)$$

afin d'approximer cette commande u^* . La fonction définie par ce système flou est

$$f_s(x) = \sum_{i=1}^m \xi^i(x) \theta^i, \quad (2.19)$$

dans laquelle θ^i est tel $\Theta^i(\theta^i) = 1$. L'erreur minimale d'approximation w est obtenue pour les paramètres θ^{i*} . Considérons alors la loi d'adaptation des paramètres θ^i , $i=1$ à m ,

$$\dot{\theta}^i = \gamma \xi^i(x) E^T P_n, \quad (2.20)$$

où γ est une constante positive et P_n est la dernière colonne de la matrice définie symétrique positive P satisfaisant la condition

$$\Lambda^T P + P \Lambda = -Q, \quad (2.21)$$

avec $Q > 0$ (choisie arbitrairement par l'utilisateur) et

$$\Lambda = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \\ 0 & 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_n \end{bmatrix}. \quad (2.22)$$

Cette loi assure pour la fonction de Lyapounov

$$V = \frac{1}{2} E^T P E + \frac{b}{2\gamma} \Phi^T \Phi, \quad (2.23)$$

où $\Phi = [\Phi_1, \dots, \Phi_m]^T$ avec $\Phi_i = \theta^{i*} - \theta^i$,

$$\dot{V} = -\frac{1}{2} E^T Q E - E^T P B w, \quad (2.24)$$

avec $B = [0, \dots, 0, b]^T \in \mathfrak{R}^n$. Si l'erreur minimale d'approximation $w \in L_2$, c'est-à-dire si w est tel que

$$\int_0^{\infty} |w(t)|^2 dt < \infty \quad (2.25)$$

alors l'erreur entre la sortie désirée et la sortie du système converge vers 0. Malheureusement, cette condition peut difficilement être vérifiée a priori et par conséquent la loi d'adaptation proposée est en général validée qu'après simulations.

2.3.2 Remarques

Cette commande floue adaptative permet d'intégrer l'analyse d'un expert lors de l'initialisation des paramètres θ^i . On augmente ainsi la vitesse d'apprentissage du système flou. Néanmoins, en considérant que les conclusions sont indépendantes les unes des autres, on perd la possibilité d'analyser le système. En effet, une fois adaptée, les conclusions de la base de règles ne sont plus liées à des termes linguistiques tels que *Grand Positif* ou *Petit Négatif*.

3 Analyse d'un système flou

3.1 Présentation

L'objectif de cette partie est d'étendre les résultats précédents à un système flou dans lequel l'agrégation des règles floues est obtenue par l'application d'une T-conorme. De plus, les influences de l'ensemble des paramètres d'un système flou (opérateurs utilisés, interface de fuzzification) sur le comportement entrée-sortie du système seront analysées.

Considérons un système flou à m règles floues et à r conclusions distinctes ($r \leq m$) de la forme :

$$\begin{array}{l}
 \left. \begin{array}{l} R^{1,1} \\ \vdots \\ R^{1,r_1} \end{array} \right\} \rightarrow \text{Conclusion } Y^1 \\
 \vdots \\
 \left. \begin{array}{l} R^{i,1} \\ \vdots \\ R^{i,r_i} \end{array} \right\} \rightarrow \text{Conclusion } Y^i \\
 \vdots \\
 \left. \begin{array}{l} R^{r,1} \\ \vdots \\ R^{r,r_r} \end{array} \right\} \rightarrow \text{Conclusion } Y^r
 \end{array} \tag{2.26}$$

avec la condition :

$$\sum_{i=1}^r r_i = m . \tag{2.27}$$

3.2 Fuzzification par la méthode du singleton.

3.2.1 Analyse

Le choix de la méthode du singleton pour la méthode de fuzzification simplifie de façon notable la détermination de la fonction d'appartenance caractérisant le sous-ensemble flou Y de la sortie du système flou. En effet, il est obtenu par l'application de l'opérateur de

projection floue (i.e. la fonction *sup*) sur l'univers de discours de la sortie. On a par conséquent :

$$\mu_Y(y) = \sup_{x \in U} (\mu_{X^0}(x) \wedge_1 \mu_R(x, y)), \quad (2.28)$$

où X^0 est le sous-ensemble flou obtenu par fuzzification de l'entrée $x^0 = [x_1^0, \dots, x_n^0] \in U$ et R est la relation globale liant les sous-ensembles flous d'entrées X_k^{ij} aux sorties Y^i . En considérant les propriétés des T-normes (opérateur d'intersection \wedge), la fonction d'appartenance μ_Y devient :

$$\mu_Y(y) = \mu_R(x^0, y) = \mu_{R^{1,1}}(x^0, y) \vee \dots \vee \mu_{R^{r,r}}(x^0, y), \quad (2.29)$$

où \vee est une T-conorme utilisée pour agréger les m règles R^{ij} . Celles-ci sont caractérisées par :

$$\mu_{R^{i,j}}(x, y) = \mu_{Y^i}(y) \wedge_3 (\mu_{X_1^{i,j}}(x_1) \wedge_2 \dots \wedge_2 \mu_{X_n^{i,j}}(x_n)), \quad (2.30)$$

où \wedge_2 et \wedge_3 (T-normes) sont respectivement un opérateur de conjonction et une implication floue. Si les sous-ensembles flous de sortie Y^i sont des singletons, cette expression devient

$$\begin{aligned} \mu_{R^{i,j}}(x, y) &= \mu_{X_1^{i,j}}(x_1) \wedge_2 \dots \wedge_2 \mu_{X_n^{i,j}}(x_n) \text{ si } y = y^i, \\ &= 0 \text{ si } y \neq y^i. \end{aligned} \quad (2.31)$$

La défuzzification barycentrique donne alors

$$y = \sum_{i=1}^r y^i \xi^i(x), \quad (2.32)$$

avec

$$\xi^i(x) = \frac{\mu_Y(y^i)}{\sum_{i=1}^r \mu_Y(y^i)} = \frac{\bigvee_{j=1}^{r_i} (\mu_{X_1^{i,j}}(x_1) \wedge_2 \dots \wedge_2 \mu_{X_n^{i,j}}(x_n))}{\sum_{i=1}^r \bigvee_{j=1}^{r_i} (\mu_{X_1^{i,j}}(x_1) \wedge_2 \dots \wedge_2 \mu_{X_n^{i,j}}(x_n))}. \quad (2.33)$$

Exemple :

Si les opérateurs sont des opérateurs probabilistes (i.e. $x \wedge y = xy$ et $x \vee y = x + y - xy$), on obtient :

$$\mu_Y(y^i) = \sum_{p=1}^{r_i} \left(\prod_{j=1}^n \mu_{x_j^{i,p}}(x_j) - \sum_{\substack{q=2 \\ q>p}}^{r_i} \left(\prod_{j=1}^n \mu_{x_j^{i,q}}(x_j) - \left(\dots - \prod_{j=1}^n \mu_{x_j^{i,r_i}}(x_j) \right) \right) \right) \quad (2.34)$$

Pour les opérateurs de Zadeh (i.e. $x \wedge y = \min(x, y)$ et $x \vee y = \max(x, y)$), cette équation devient :

$$\mu_Y(y^i) = \max \left(\min_{j=1 \text{ à } n} \left(\mu_{x_j^{i,1}}(x_j) \right), \dots, \min_{j=1 \text{ à } n} \left(\mu_{x_j^{i,r_i}}(x_j) \right) \right) \quad (2.35)$$

Théorème

Soit Z l'ensemble des fonctions de la forme de l'équation (2.32) d'un ensemble compact $U = U_1 \times \dots \times U_n$, alors toute fonction f continue de U peut être approximée avec la précision désirée ε par un système flou S à m règles de Z .

La démonstration de ce théorème est donnée dans l'annexe 2.

3.2.2 Commande floue adaptative

La loi d'adaptation des paramètres $y^i, i = 1 \text{ à } r$ proposée par Wang

$$\dot{y}^i = \gamma \xi^i(x) E^T P_n \quad (2.36)$$

reste valable et assure pour un système flou caractérisé par l'équation (2.32) et une partition pas trop grossière de l'ensemble compact U (i.e. une erreur minimale d'approximation w faible) la convergence de l'erreur vers 0.

Exemple :

Si nous considérons le système du pendule inversé commandé par un système flou utilisant la base à 9 règles suivante (où N : Négatif, Z : Zéro, P : Positif, GN : Grand Négatif, PN : Petit Négatif, PP : Petit Positif, GP : Grand Positif) :

$\Delta e \backslash e$	N	Z	P
N	GN	PN	Z
Z	PN	Z	PP
P	Z	PP	GP

Tableau 2.1 : Base de règles pour la commande du pendule inversé.

La partition de l'ensemble $U = U_1 \times U_2$ est réalisée par trois sous-ensembles flous sur les domaines de l'erreur $e = \theta - \theta_d$ et de la variation de l'erreur Δe .

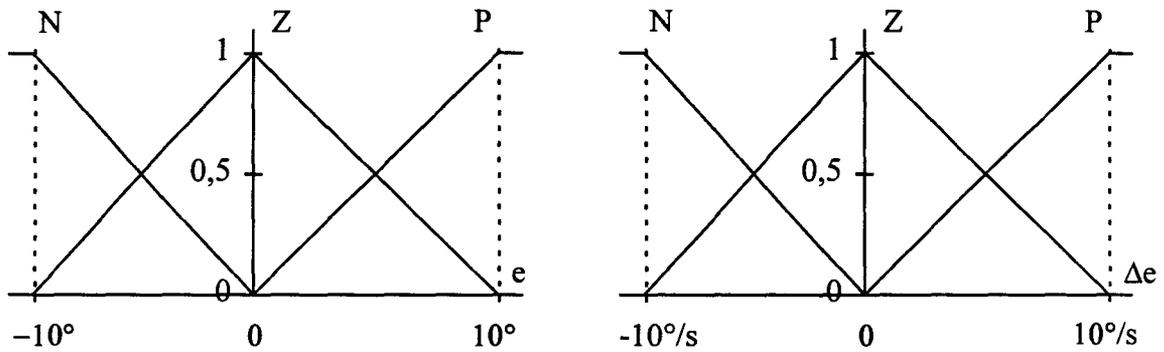


Figure 2.6: Partition des ensembles U_i .

Pour les valeurs initiales $GP = -GN = 1$, $PP = -PN = 0,5$, $Z = 0$, les figures suivantes présentent la comparaison entre la commande adaptative de Wang et la commande floue utilisant une approche logique, pour différents opérateurs duaux \wedge et \vee .

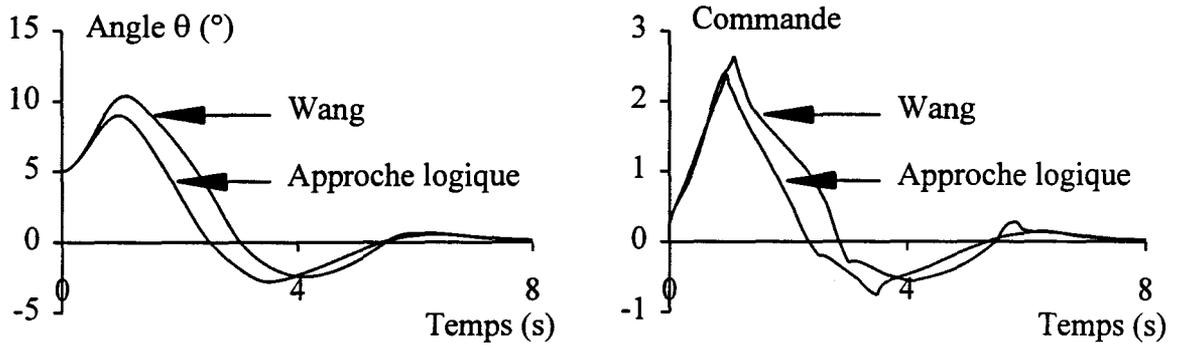


Figure 2.7 : Opérateurs de Zadeh.

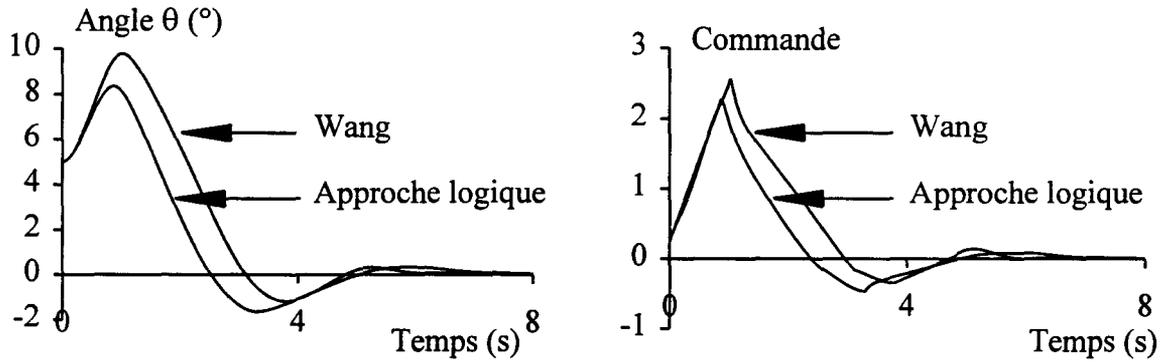


Figure 2.8 : Opérateurs probabilistes.

La réalisation d'une commande floue adaptative par une approche logique offre les mêmes avantages que celle développée par Wang. En effet, elle permet d'intégrer l'analyse d'un expert et assure la convergence de l'erreur de poursuite vers 0. De plus elle permet de conserver la forme initiale de la base de règle. Ainsi, l'analyse qualitative des performances de la commande réalisée reste possible même pour une base de règles complexe.

3.3 Influence de la méthode de fuzzification

Considérons le cas d'une méthode de fuzzification probabiliste (qui englobe la méthode du singleton). Le sous-ensemble flou $X^0(x)$ d'une mesure x^0 est caractérisé par la fonction d'appartenance

$$\mu_{x^0}(x) = \max\left(0, 1 - \frac{|x - x^0|}{\varepsilon}\right), \tag{2.37}$$

et dont sa représentation graphique sur l'univers de discours U_k est

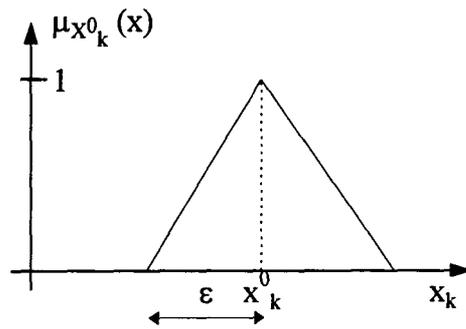


Figure 2.9 : Fuzzification probabiliste.

Afin d'analyser l'influence du paramètre ε sur la fonction f_s du système flou, il est nécessaire d'étudier l'intégralité de la chaîne des opérations effectuées. La sortie du système flou est obtenue par défuzzification de l'union des règles R^{ij} . La règle R^{ij} est caractérisée par

$$\begin{aligned}
 \mu_{R^{i,j}}(y) &= \sup_{x \in U} (\mu_{X^0}(x) \wedge_1 (\mu_{X^{i,j}}(x) \wedge_3 \mu_{Y^i}(y))) \\
 &= \mu_{Y^i}(y) \wedge_3 \sup_{x \in U} (\mu_{X^0}(x) \wedge_1 \mu_{X^{i,j}}(x)) \\
 &= \mu_{Y^i}(y) \wedge_3 \left(\bigwedge_{k=1 \text{ à } n} \sup_{x_k \in U_k} (\mu_{X_k^0}(x_k) \wedge_1 \mu_{X_k^{i,j}}(x_k)) \right).
 \end{aligned}
 \tag{2.38}$$

Par conséquent, il faut étudier l'influence de ε sur l'expression

$$\sup_{x_k \in U_k} (\mu_{X_k^0}(x_k) \wedge_1 \mu_{X_k^{i,j}}(x_k)).
 \tag{2.39}$$

Les fonctions d'appartenance des sous-ensembles flous $X_k^{i,j}$ ont généralement une forme trapézoïdale telle que celle présentée sur la figure ci-dessous.

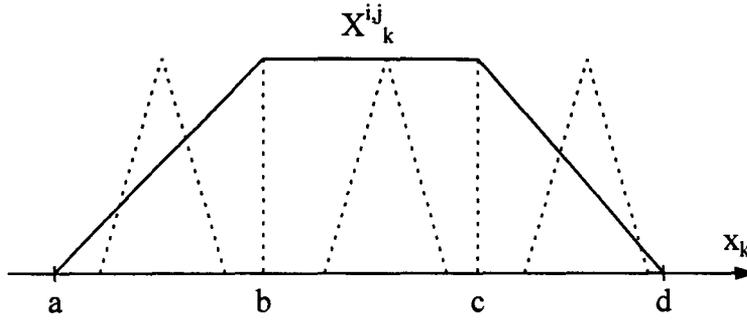


Figure 2.10 : Forme usuelle d'une fonction d'appartenance du sous-ensemble flou X_k^{ij} .

En dehors de l'intervalle $[b ; c]$, l'appartenance de x_k au sous-ensemble flou X_k^{ij} décroît de façon continue et s'annule en a et d . On recherche à déterminer l'évolution de l'expression (2.39) en fonction du paramètre ε de la fuzzification probabiliste.

Pour ce type de fonction d'appartenance, si x_k^0 est dans l'intervalle $[b ; c]$ alors

$$\forall \varepsilon, g(x_k^0) = \sup_{x_k \in U_k} (\mu_{x_k^0}(x_k) \wedge_1 \mu_{X_k^{i,j}}(x_k)) = 1. \tag{2.40}$$

3.3.2 Utilisation de la fonction min

Si la T-norme \wedge_1 est la fonction *min*, on obtient dans le domaine $[a ; b]$ (respectivement $[c ; d]$)

$$g(x_k^0) = \sup_{x_k \in U_k} \min(\mu_{x_k^0}(x_k), \mu_{X_k^{i,j}}(x_k)) = \frac{x_k^0 + \varepsilon - a}{\varepsilon + (b - a)} \tag{2.41}$$

$$\left(\text{resp. } \frac{\varepsilon + d - x_k^0}{\varepsilon + (d - c)} \right).$$

Ainsi, l'utilisation de la fonction *min* dans la méthode de fuzzification probabiliste permet de limiter l'influence de la mesure x^0 (et donc, de manière indirecte, diminue la sensibilité aux bruits) sur la totalité des sous-ensembles flou X_k^{ij} , $i = 1$ à r , $j = 1$ à r_i , $k = 1$ à n . Par conséquent, l'influence de x^0 sur les règles R^{ij} , caractérisées par l'équation

$$\mu_{R^{i,j}}(y) = \mu_{Y^i}(y) \wedge_3 \left(\bigwedge_{k=1 \text{ à } n} \sup_{x_k \in U_k} (\mu_{x_k^0}(x_k) \wedge_1 \mu_{X_k^{i,j}}(x_k)) \right), \tag{2.42}$$

est aussi réduite. Il en est de même pour la sortie du système flou S obtenue par la défuzzification de l'agrégation de ces règles.

Les figures suivantes présentent les évolutions de l'expression (2.39) et de $f_s(x)$ (définie par l'équation (2.44)) pour différentes valeurs de ε dans le cas d'un système flou monovarié décrit par deux règles R^i

$$R^i : \text{Si } x \text{ est } X^i \text{ ALORS } y \text{ est } Y^i. \quad (2.43)$$

où les Y^i sont des singletons flous distincts ($y^1 = 0$ et $y^2 = 1$) et les sous-ensembles flous X^i sont définis sur la figure ci-dessous.

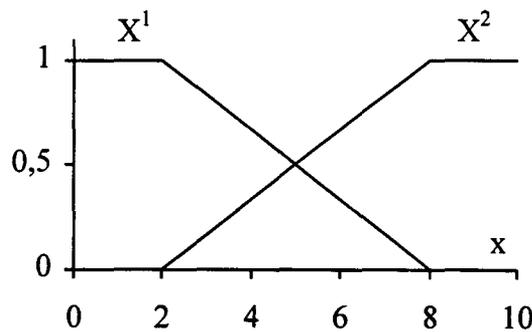


Figure 2.11 : Définition des sous-ensembles flous X^i .

L'équation du système flou est alors

$$f_s(x) = \frac{\mu_2}{\mu_1 + \mu_2}, \quad (2.44)$$

dans laquelle,

$$\mu_i = \sup_{x \in [0;10]} (\mu_{X^0}(x) \wedge_1 \mu_{X^i}(x)). \quad (2.45)$$

On prend pour valeurs de ε 25% et 50% du domaine de chevauchement des sous-ensembles flous X^1 et X^2 (i.e. $\varepsilon = 3/2$ ou 3).

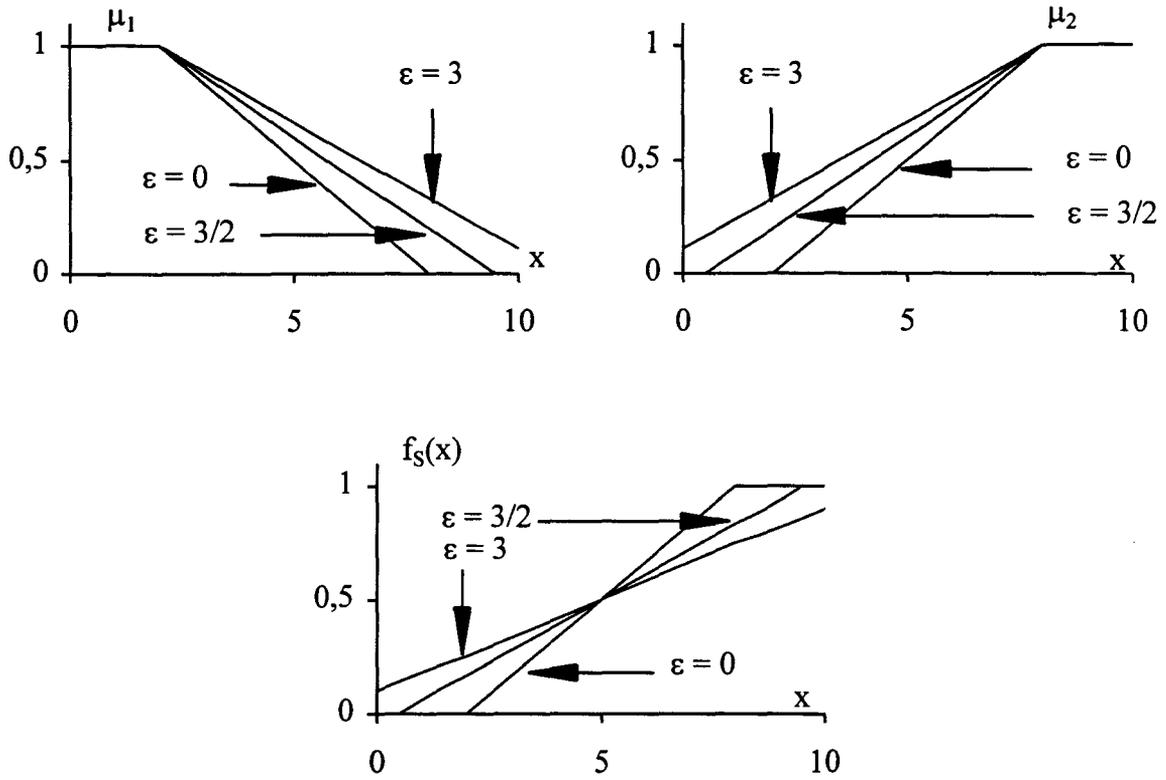


Figure 2.12 : Influence de ϵ pour la fonction *min*.

3.3.2 Utilisation de la fonction produit

Si la T-norme \wedge_1 est la fonction *produit*, alors dans le domaine $[a ; b]$ (respectivement $[c ; c]$), si $x_k^0 \geq 2b - a - \epsilon$ (resp. $x_k^0 \leq 2c + \epsilon - d$)

$$g(x_k^0) = \sup_{x_k \in U_k} (\mu_{x_k^0}(x_k) \mu_{x_k^{i,j}}(x_k)) = \mu_{x_k^0}(b) \quad (2.46)$$

(resp. $\mu_{x_k^0}(c)$),

sinon si $x_k^0 \geq a + \epsilon$ (resp. $x_k^0 \leq d - \epsilon$) on obtient

$$g(x_k^0) = \sup_{x_k \in U_k} (\mu_{x_k^0}(x_k) \mu_{x_k^{i,j}}(x_k)) = \mu_{x_k^{i,j}}(x_k^0), \quad (2.47)$$

et dans le cas contraire :

$$g(x_k^0) = \sup_{x_i \in \dots U_i} (\mu_{x_{0,i}}(x_i) \mu_{x_i^*}(x_i)) = \frac{(x_i^j + \varepsilon - a)^2}{4\varepsilon(b-a)} \quad (2.48)$$

$$\left(\text{resp. } \frac{(d + \varepsilon - x_i^j)^2}{4\varepsilon(d-c)} \right).$$

Ainsi, l'influence de ε sur le sous-ensemble flou X_k^{ij} dépend directement de la mesure x_k^0 : plus cette mesure tend à ne plus appartenir au sous-ensemble flou considéré plus celui-ci sera insensible à x_k^0 (et, de façon indirecte, aux bruits).

Les figures suivantes présentent les évolutions de l'expression (2.39) et de $f_S(x)$ pour différentes valeurs de ε dans le cas du système flou monovarié décrit plus haut.

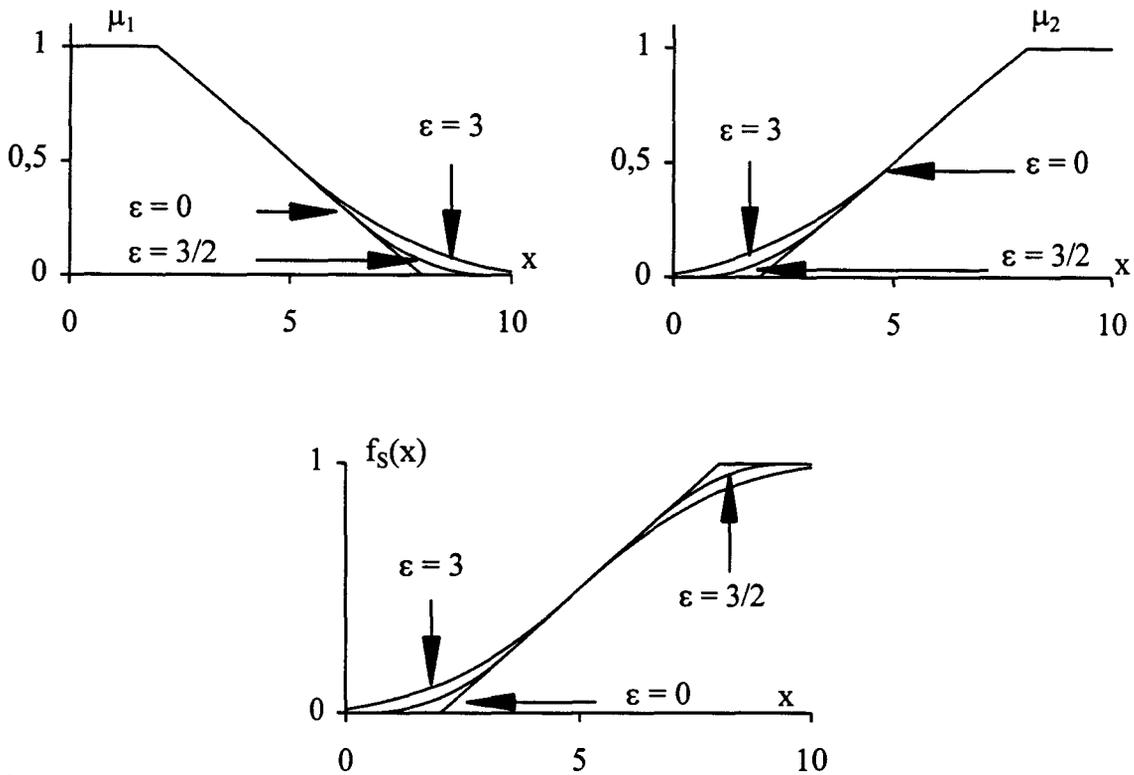


Figure 2.13 : Influence de ε pour la fonction produit.

3.3.3 Remarque

La fuzzification probabiliste limite les capacités d'approximation d'un système flou. En effet, l'insensibilité de $f_S(x)$ aux bruits de mesure est obtenue en diminuant l'influence de x^0 sur les règles R^{ij} . Ainsi, l'hypersurface générée par le système flou $f_S(x)$ tend à se niveler avec l'augmentation de ε et on obtient pour limite l'hyperplan caractérisé par la relation

$$\lim_{\varepsilon \rightarrow \infty} f_{\mathcal{S}}(x) = \frac{\sum_{i=1}^r y^i}{r}. \quad (2.49)$$

Exemple :

Les figures ci-dessous présentent, pour le système flou à 9 règles floues utilisé pour la commande du pendule inversé, l'aspect de la surface $f_{\mathcal{S}}(x)$ pour une fuzzification probabiliste avec $\varepsilon_1 = \varepsilon_2 = 0$ (méthode du singleton) et $\varepsilon_1 = 7,5^\circ$ et $\varepsilon_2 = 7,5^\circ/s$.

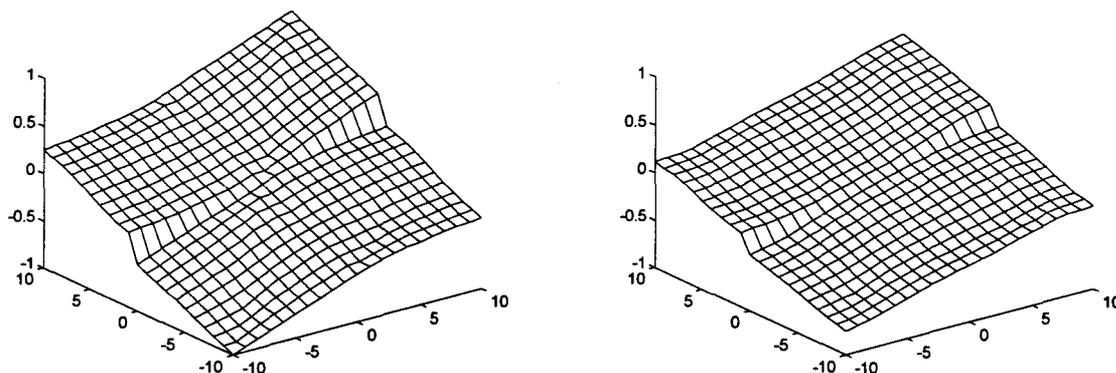


Figure 2.14 : Surface $f_{\mathcal{S}}(x)$.

4 Conclusions

L'étude d'un système flou par une approche non-linéaire a permis de démontrer les propriétés d'approximation d'un tel système. Par conséquent, l'utilisation de la logique floue dans la commande de systèmes est justifiée.

Il a été montré qu'à partir de l'analyse qualitative d'un expert, on peut construire une commande floue adaptative assurant la stabilité d'un système non linéaire. De plus, nos travaux ont mis en évidence l'intérêt de l'utilisation d'une T-conorme afin d'agrèger les règles floues. En effet, cela permet à l'opérateur de garder la maîtrise du système de commande.

Enfin, l'analyse de l'influence d'une fuzzification probabiliste a montré que cette méthode permet de diminuer l'influence des bruits de mesure sur la sortie du système flou en diminuant l'influence de la mesure elle même. De ce fait, cette insensibilité aux bruits se fait au détriment des capacités d'approximation du système flou.

Dans des problèmes de commande de processus, l'expert fournissant la base de règles est le plus souvent capable de fournir des conclusions autres que des actions qualitatives telle *la sortie est Positive*. En particulier, il peut fournir des lois de commande à appliquer pour des configurations données du système.

Dans le troisième chapitre nous étudierons les apports de la logique floue dans une approche multi-modèle de la commande d'un système complexe.

Chapitre 3 : COMMANDE FLOUE MULTI-MODELE

1 Introduction

Les systèmes réels sont le plus souvent non linéaires, non autonomes et même mal définis. Nous ne connaissons ces systèmes que par des modèles plus ou moins précis. Néanmoins, c'est à partir d'un modèle particulier (mathématique ou autre) que nous construisons une commande permettant d'obtenir les résultats désirés (figure 3.1). Par conséquent, cette commande, lorsqu'elle est appliquée au système réel, doit se révéler suffisamment robuste pour compenser les imperfections du modèle. La robustesse d'une commande se faisant au détriment de ses performances, le modèle doit être le plus proche possible du système. Pourtant, il peut devenir difficile (et même impossible) d'obtenir l'expression analytique de la commande si le modèle est trop complexe. Ainsi, la mise en place d'une commande doit prendre en compte et optimiser le triplet Robustesse, Performance, Simplicité.

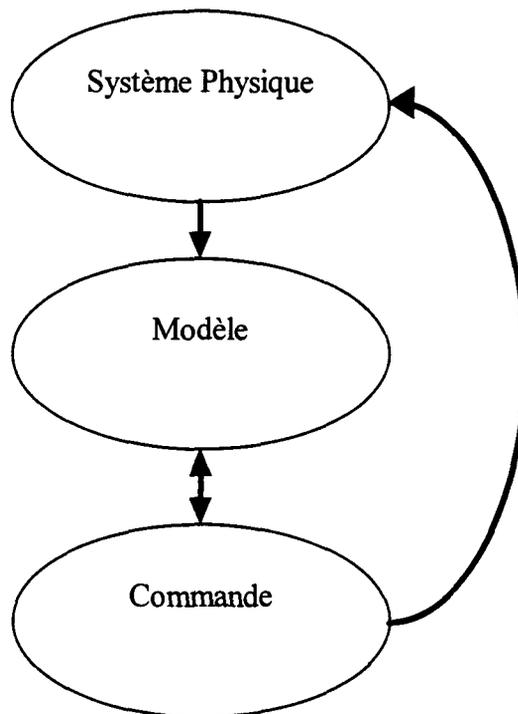


Figure 3.1 : Relation entre système physique, modèle et commande.

Dans ce chapitre, nous présentons une méthode de commande basée sur une approche multi-modèle. Celle-ci consiste à considérer non pas un unique modèle du système étudié mais un ensemble de modèles simples pour lesquels des commandes sont définies. Un algorithme est alors utilisé pour fournir une représentation du système et permettre de réaliser une commande globale à partir des commandes élémentaires.

2 Principe de la commande multi-modèle

2.1 Utilisation d'un modèle linéarisé

L'étude des propriétés d'un système non linéaire débute souvent par l'analyse d'un modèle linéarisé autour d'un point de fonctionnement de l'espace d'état. En effet, on peut déduire de cette étude certains comportements du système initial.

La réalisation d'une commande à partir d'un modèle linéarisé offre de nombreux avantages (principalement la simplicité de mise en oeuvre). Il faut toutefois émettre des réserves sur la validité d'une telle commande, même dans un voisinage du point de fonctionnement si le système est fortement non linéaire. Par conséquent, il est nécessaire de vérifier la robustesse de cette commande vis-à-vis des imperfections du modèle. De plus, il est important de définir le domaine de l'espace d'état dans lequel les propriétés de la commande restent valides.

2.2 Approche multi-modèle

Afin d'accroître le domaine de validité de la commande, il peut être utile de définir plusieurs modèles du système autour de différents points de fonctionnement (figure 3.2).

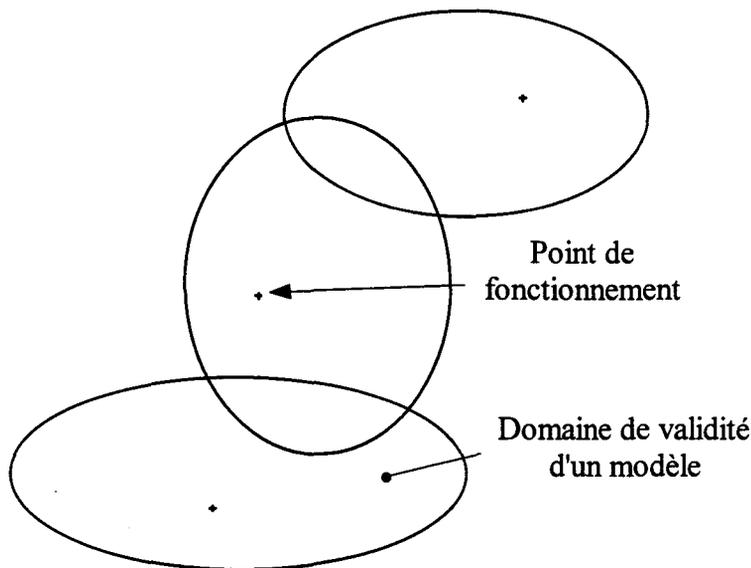


Figure 3.2 : Utilisation de plusieurs modèles locaux.

Une commande particulière satisfaisant les critères désirés par l'utilisateur est associée à chaque modèle. Dès lors, il est nécessaire d'établir une politique de passage entre ces différentes commandes.

Une stratégie de passage usuelle consiste à déterminer le plus proche point de fonctionnement de l'état du système. On applique alors la commande associée au modèle correspondant. La mise en place d'un gain variable permet d'éviter les à-coups lors de la commutation entre deux commandes. La figure suivante présente ces deux méthodes de commutation.

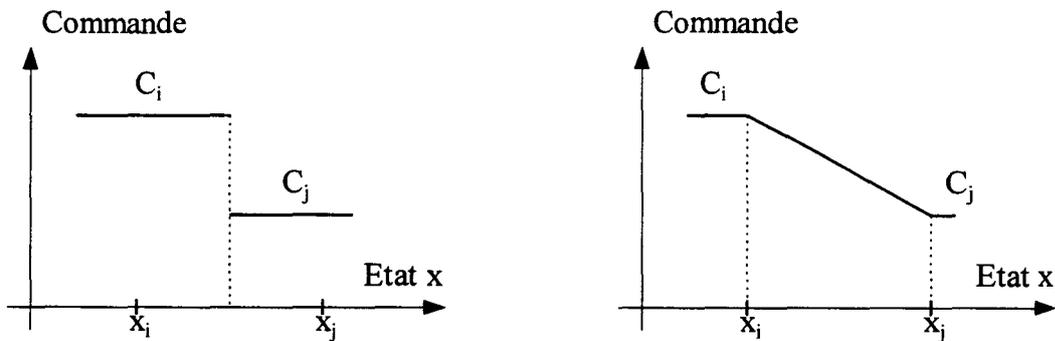


Figure 3.3 : Méthodes de commutation usuelles dans une approche multi-modèle.

La logique floue peut être considérée comme une méthode d'interpolation entre les conclusions apportées par l'observation d'un même fait traité par des règles différentes. Il nous a semblé intéressant d'associer l'aspect traitement de plusieurs sources d'informations de la logique floue avec l'approche multi-modèle.

3 Commande floue multi-modèle

3.1 Historique de la commande multi-modèle

3.1.1 Modélisation floue de Takagi & Sugeno

Les commandes floues traditionnelles sont centrées autour d'une base de règles ayant la forme suivante :

$$R^i : \text{SI } x_1 \text{ est } X_1^i \text{ ET } \dots \text{ ET } x_n \text{ est } X_n^i \text{ ALORS } y \text{ est } Y^i. \quad (3.1)$$

Dans cette équation, $x = [x_1, \dots, x_n]^T \in \mathfrak{R}^n$ et y sont respectivement le vecteur de mesure et la sortie du système flou ; X_j^i est le sous-ensemble flou sur l'univers de discours de la variable x_j et Y^i est le sous-ensemble flou sur l'univers de discours de la sortie pour la règle R^i .

Pour traiter un problème, la première étape consiste à identifier le cas qui se présente à nous. Cette étape correspond au calcul de l'appartenance du fait observé aux classes déterminées par l'expert (prémises des règles floues). De plus, pour un problème parfaitement défini, la conclusion à prendre est le plus souvent dépourvue d'ambiguïtés. C'est pourquoi, on peut envisager de remplacer les sous-ensembles flous Y^i des règles floues R^i par des conclusions non floues. Cette idée est appliquée dans les commandes floues traditionnelles où les Y^i sont des singletons flous, c'est-à-dire des sous-ensembles représentatifs de faits précis.

Takagi et Sugeno [Tak 1985] furent les premiers à modifier de façon notable la règle (3.1) afin de parvenir à une modélisation floue d'un système n'utilisant pas dans les conclusions de termes grammaticaux. La forme des règles floues utilisées est :

$$R^i : \text{SI } x_1 \text{ est } X_1^i \text{ ET } \dots \text{ ET } x_n \text{ est } X_n^i \text{ ALORS } y \text{ est } f_i(x), \quad (3.2)$$

où $f_i(x)$ représente un modèle simplifié du processus à modéliser pour un point particulier de l'espace d'état. La fonction f_i , $i = 1$ à m , est définie par :

$$f_i(x) = p_0^i + \sum_{j=1}^n p_j^i x_j. \quad (3.3)$$

Ainsi, après l'application de l'opérateur de défuzzification, le système réel est décomposé en une combinaison des f_i dans les hypercubes définis par les prémisses des règles floues R^i . Par exemple, pour m règles floues, on obtient une approximation du système $\dot{x} = f(x)$:

$$\hat{f}(x) = \frac{\sum_{i=1}^m \mu_i(x) f_i(x)}{\sum_{i=1}^m \mu_i(x)} \quad (3.4)$$

Dans cette équation, les $\mu_i(x)$ représentent les valeurs de l'appartenance du vecteur état x aux prémisses x_1 est X_1^i ET ... ET x_n est X_n^i et sont définies par :

$$\begin{aligned} \mu_i(x) &= \mu_{x_1 \text{ est } X_1^i \text{ ET } \dots \text{ ET } x_n \text{ est } X_n^i}(x) \\ &= \prod_{j=1}^n \mu_{x_j \text{ est } X_j^i}(x_j) \end{aligned} \quad (3.5)$$

Une fois les fonctions d'appartenance des faits x_j est X_j^i définies (i.e. la partition de l'espace d'état effectuée), une phase d'identification est nécessaire afin de déterminer les coefficients p_j^i des fonctions f_i .

Il est possible d'utiliser cette méthode de modélisation pour la réalisation d'une commande floue. Dans ce cas, les fonctions $f_i(x)$ représentent les commandes à mettre en oeuvre pour des points de fonctionnement particuliers.

3.1.2 Commande multi-modèle de Ishigame

En 1993, Ishigame et al. [Ish 1993] utilisèrent le principe de la modélisation floue exposée plus haut afin de commander un système électrique. Les règles floues utilisées pour modéliser le système $\dot{x} = f(x) + BU$ sont du type :

$$R^i : \text{SI } x_1 \text{ est } X_1^i \text{ ET } \dots \text{ ET } x_n \text{ est } X_n^i \text{ ALORS } \dot{x} = A_i x + BU_i \quad (3.6)$$

A partir d'un ensemble de règles de cette forme une approximation du système est donnée par :

$$\dot{x} = \frac{\sum_{i=1}^m \mu_i(x)(A_i x + B U_i)}{\sum_{i=1}^m \mu_i(x)}, \quad (3.7)$$

où les μ_i sont identiques à ceux définis dans l'équation (3.5).

Pour chaque modèle $\dot{x} = A_i x + B U_i$, on détermine une commande particulière U_i permettant d'obtenir les résultats souhaités pour le modèle. La commande du système réel $\dot{x} = f(x) + B U$ est alors obtenue de la façon suivante :

$$U(t) = \frac{\sum_{i=1}^m \mu_i U_i(t)}{\sum_{i=1}^m \mu_i}. \quad (3.8)$$

3.1.3 Commande basée sur une modélisation floue de Johansen

Récemment, Johansen [Joh 1994] a développé une commande floue basée sur la modélisation floue proposée par Takagi et Sugeno. Les modèles choisis sont des modèles ARX et les règles utilisées sont de la forme :

$$R^i : \text{SI } z(t) \text{ est } Z_i \text{ ALORS } y(t+d) = \sum_{j=0}^{\eta_y} A_{j,i} y(t-j) + \sum_{j=0}^{\eta_u} B_{j,i} u(t-j) + C_i, \quad (3.9)$$

avec z l'état du système, Z_i la zone de fonctionnement du modèle M_i , $y = [y_1, \dots, y_p]^T$ la sortie du processus, $u = [u_1, \dots, u_r]^T$ le vecteur de commande avec $r \geq p$, η_y et η_u deux entiers positifs, $A_{i,j}$, $B_{i,j}$ et C_i des matrices et $y(t+d) = [y_1(t+d_1), \dots, y_p(t+d_p)]^T$ où les d_i sont les retards du système. Les différents paramètres de ces modèles ARX doivent être identifiés dans leurs zones de fonctionnement respectives Z_i .

La valeur d'appartenance du système à une zone de fonctionnement Z_i , $\mu_{z \text{ est } Z_i}$, est utilisée afin de déterminer un modèle global du système.

$$y(t+d) = \sum_{i=1}^m \left(\sum_{j=0}^{\eta_y} A_{j,i} y(t-j) + \sum_{j=0}^{\eta_u} B_{j,i} u(t-j) + C_i \right) \omega_i$$

$$\omega_i = \frac{\mu_{z \text{ est } Z_i}}{\sum_{j=1}^m \mu_{z \text{ est } Z_j}}, \quad (3.10)$$

où m est le nombre de modèles utilisés. Cette équation peut s'écrire de façon à obtenir un modèle ARX global pour le système.

$$y(t+d) = \sum_{j=0}^{\eta_y} A_j y(t-j) + \sum_{j=0}^{\eta_u} B_j u(t-j) + C + v(t+d). \quad (3.11)$$

Ici, $v(t+d)$ représente les différentes erreurs de modélisation et/ou les perturbations externes, telles que le bruit sur les mesures. On a de plus :

$$A_j = \sum_{i=1}^m A_{j,i} \omega_i,$$

$$B_j = \sum_{i=1}^m B_{j,i} \omega_i, \quad (3.12)$$

$$C = \sum_{i=1}^m C_i \omega_i.$$

A partir de cette représentation globale, il est possible de déterminer une commande permettant d'obtenir les résultats souhaités en rejetant l'influence de la perturbation $v(t+d)$.

3.1.4 Conclusions

L'approche multi-modèle est un sujet de recherche de plus en plus exploré. Les recherches actuelles utilisent un ensemble de modèles locaux linéaires. Deux axes principaux se dégagent.

Le premier est basé sur les études préliminaires de Takagi et Sugeno. Il consiste à définir une approximation du système par une combinaison non linéaire, obtenue par les propriétés d'interpolation de la logique floue, de modèles linéaires représentant le système dans des domaines particuliers de l'espace d'état. Une fois ce modèle global obtenu, une loi de commande est réalisée.

Le deuxième axe de recherche consiste à définir, pour chaque modèle, une commande particulière et à considérer la commande globale du système comme une interpolation de ces commandes élémentaires. A ma connaissance, seul Ishigame [Ish 1993] a envisagé ce type d'approche.

La figure 3.4 donne un résumé de ces deux approches.

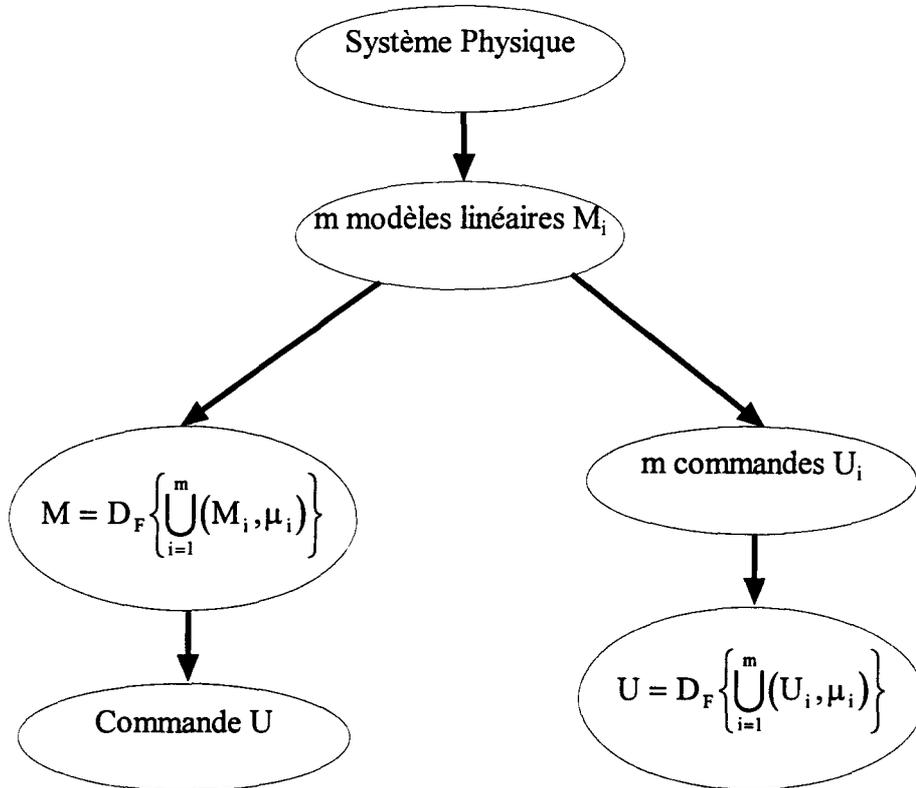


Figure 3.4 : Présentation des deux approches multi-modèles.

3.2 Approches considérées

3.2.1 Système considéré

Le système considéré dans nos travaux est caractérisé par l'équation différentielle suivante :

$$\dot{x} = f(x, u, t), \quad (3.13)$$

où $x \in \mathbb{R}^n$ est le vecteur d'état du système, $u \in \mathbb{R}^f$ est le vecteur de commande, f une fonction inconnue de l'état, du temps et de la commande.

3.2.2 Modèles du système

On considère que l'on a accès à un certain nombre de modèles M_i du système défini dans l'équation (3.13). Le i -ème modèle est décrit par l'équation différentielle :

$$M_i : \dot{x} = f_i(x, u, t). \quad (3.14)$$

où $x \in \mathcal{R}^n$ est le vecteur d'état, $u \in \mathcal{R}^r$ est le vecteur de commande, f_i une fonction connue.

3.2.3 Validité des modèles

La validité des modèles M_i définis dans (3.14) dépend de l'approche considérée. Dans un premier temps, l'étude a concerné des modèles locaux. Par conséquent, chaque modèle M_i est déterminé pour un domaine de validité particulier D_i . Par la suite, on a envisagé des domaines de validité non définis : les modèles considérés sont alors dit génériques. Dans ce cas, l'évolution du système réel est comparée à celles des modèles et l'appartenance du processus à un modèle est calculée de façon prédictive ou adaptative.

3.2.4 Résumé

La figure suivante présente de manière synthétique les axes de recherche envisagés dans nos travaux sur la commande floue multi-modèle.

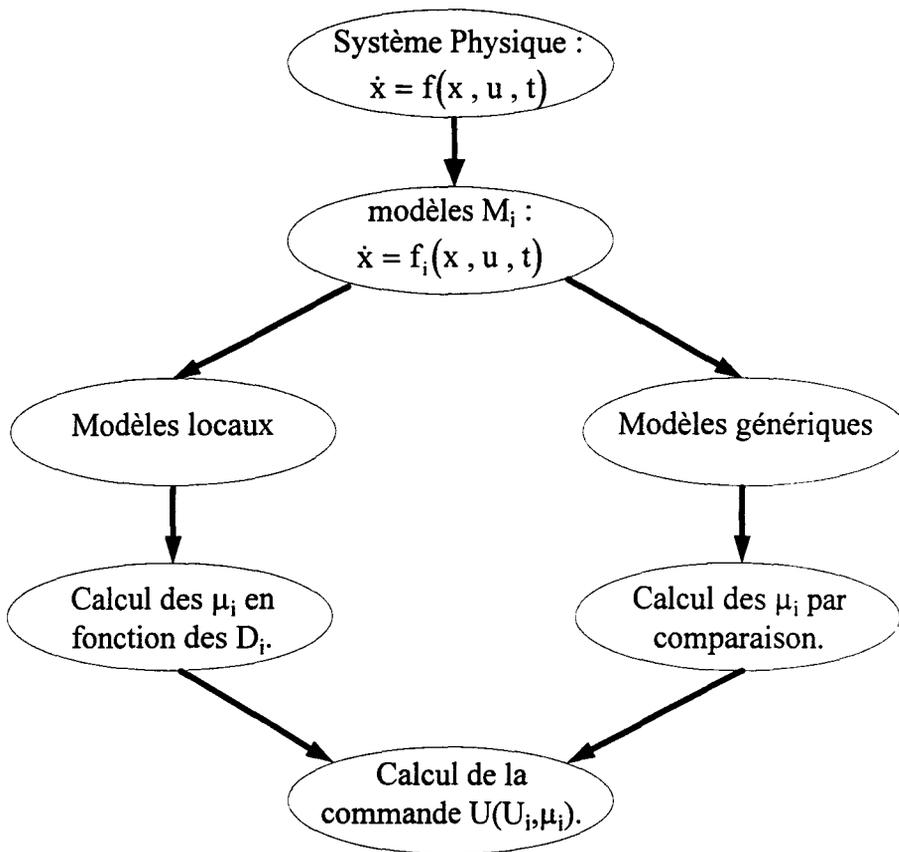


Figure 3.5 : Résumé des approches considérées.

4 Utilisation de modèles locaux

4.1 Introduction

Le principe de l'utilisation de modèles locaux dans une commande multi-modèle est naturel. En effet, on conçoit aisément que l'attention de l'utilisateur se focalise sur les points de fonctionnement pertinents du système.

Par exemple, dans le cas d'un moteur électrique, l'utilisateur désire limiter le courant au démarrage et assurer une faible erreur de régulation pour différents régimes de fonctionnement malgré les perturbations extérieures (e.g. couple résistant variable, ...). On définit alors plusieurs commandes élémentaires assurant dans des domaines particuliers (démarrage, régime permanent) les spécifications désirées. Afin d'assurer le bon fonctionnement de l'ensemble, on doit réaliser un superviseur établissant le lien entre ces différentes commandes.

Chaque modèle M_i , $i=1$ à m , du système est caractérisé par un triplet de la forme :

$$(D_i, \neg D_i, U_i) \quad (3.15)$$

où D_i et $\neg D_i$ sont respectivement les domaines de validité et de non-validité absolues du i -ème modèle et U_i est la commande dérivée de celui-ci. Dans le domaine de validité, la commande du système correspond à U_i . A l'opposé, dans le domaine $\neg D_i$, la commande U_i ne doit pas être prise en compte (e.g. pour des problèmes d'instabilité). En dehors de ces domaines particuliers, il est nécessaire de calculer les influences sur la commande du système des différentes actions U_i proposées. Ces spécifications sont présentées sur la figure suivante.

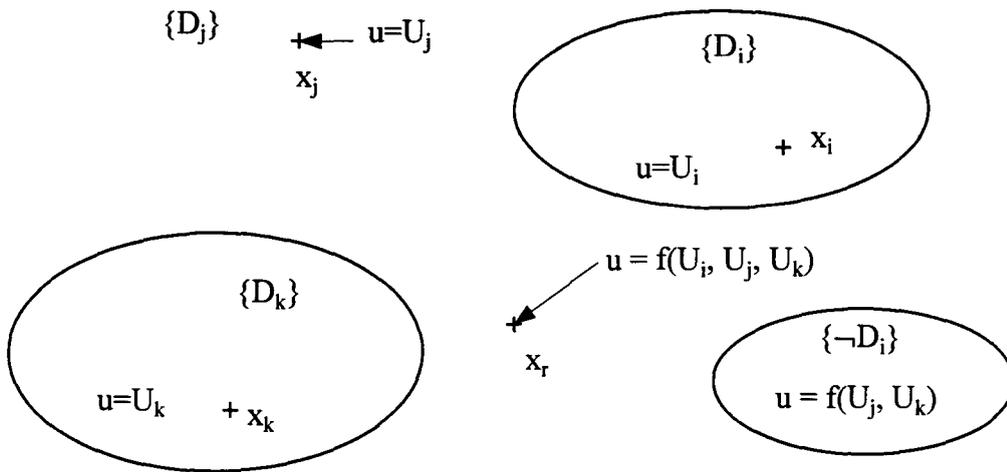


Figure 3.6 : Présentation de trois modèles d'un système (on peut noter que le domaine D_j se limite à $\{x_j\}$).

Remarque :

La définition donnée aux modèles M_i implique que notre intérêt se limite à l'action à entreprendre dans un domaine particulier D_i . Par conséquent, une commande multi-modèle peut être construite à partir d'un seul modèle (au sens classique) du système réel. C'est le cas, par exemple, si l'on considère un modèle défini autour d'un point d'équilibre du système (figure 3.7). Autour de ce point (domaine de validité D_1) une commande optimale (e.g. minimisant un critère) est mise en place; en dehors de ce domaine (domaine de validité D_2) une commande robuste est créée pour assurer la convergence de l'état du système vers ce point d'équilibre malgré les perturbations (modélisation peu précise, ...).

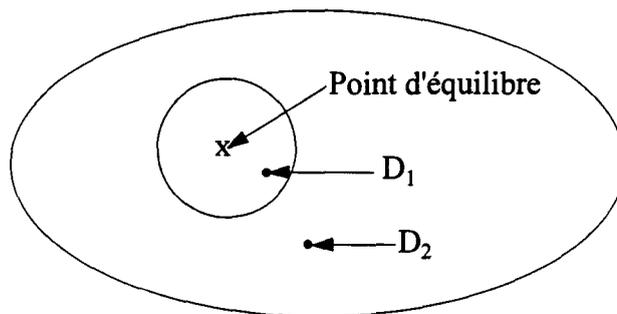


Figure 3.7 : 1 seul modèle pour une commande multi-modèle.

4.2 Commande floue multi-modèle

4.2.1 Structure

La figure suivante présente la structure de la commande floue multi-modèle proposée. Elle se compose des m modèles M_i caractérisés par les triplets $(D_i, -D_i, U_i)$ ainsi que d'un

mécanisme d'inférence. Celui-ci a pour objectif de calculer la commande à appliquer au système en fonction de l'ensemble des informations disponibles.

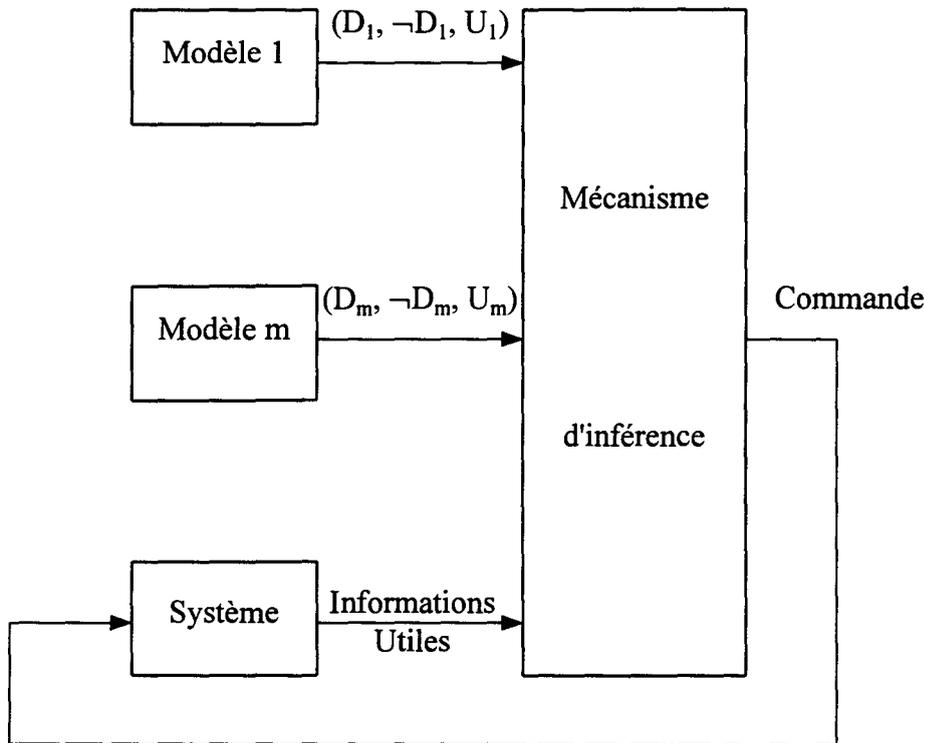


Figure 3.8 : Structure de la commande multi-modèle proposée.

4.2.2 Principe de fonctionnement

La commande floue multi-modèle utilise des règles floues R^i de la forme :

$$R^i : \text{SI le système est le } M_i \text{ ALORS la commande est } U_i. \quad (3.16)$$

Comme dans le cas de la commande floue traditionnelle, l'objectif est de définir le sous-ensemble flou U_S de l'espace de commande déduit des commandes U_i proposées par les modèles M_i , $i = 1$ à m . Un opérateur de défuzzification est alors employé pour déterminer la commande u_S que l'on applique au système.

La détermination de cette commande s'effectue en trois étapes :

- 1- Calcul des confiances dans les modèles M_i ,
- 2- Calcul du sous-ensemble flou U_S par la fusion des commandes U_i proposées,
- 3- Défuzzification de U_S .

Calcul des confiances :

Le calcul des confiances dans les modèles M_i est basée sur les distances de l'état représentatif du système (vecteur $x_r = [x_{r_1}, \dots, x_{r_{n_r}}]^T \in \mathfrak{R}^{n_r}$) aux différents domaines associés aux modèles (i.e. D_i et $\neg D_i$). Ainsi, plus cet état est éloigné d'un modèle particulier M_i (relativement aux autres) plus sa confiance t_i doit être faible. Les distances d_i et $\neg d_i$ sont obtenues de la façon suivante :

$$\begin{aligned} d_i &= \min_{x \in D_i} \|x_r - x\|, \\ \neg d_i &= \min_{x \in \neg D_i} \|x_r - x\|, \end{aligned} \tag{3.17}$$

où $\| \cdot \|$ est une norme, par exemple la norme euclidienne :

$$\|x_r - x\| = \left(\sum_{i=1}^{n_r} (x_{r_i} - x_i)^2 \right)^{\frac{1}{2}}, \tag{3.18}$$

avec $x_r = [x_{r_1}, \dots, x_{r_{n_r}}]^T$ et $x = [x_1, \dots, x_{n_r}]^T$. La figure qui suit présente un exemple de construction des distances d_i .

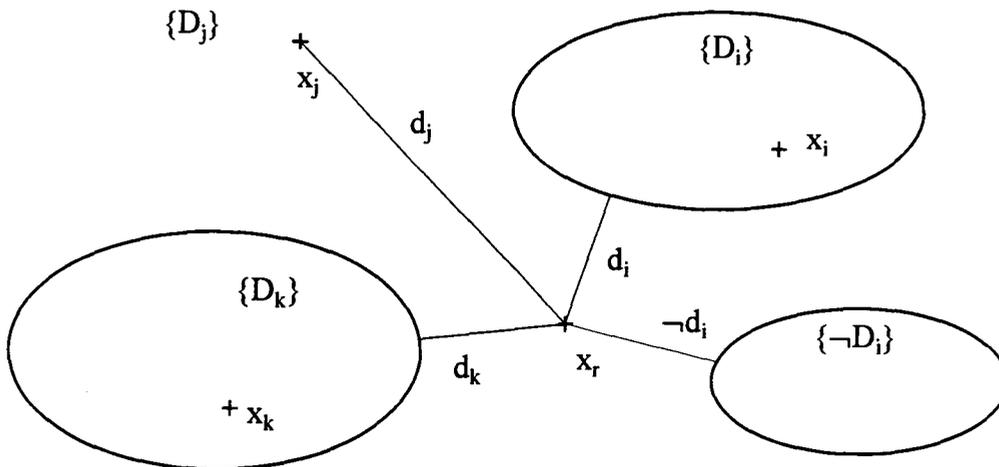


Figure 3.9 : Distance de l'état représentatif du système aux différents domaines.

Etat représentatif du système :

On appelle état représentatif du système le vecteur $x_r \in \mathfrak{R}^{n_r}$ dont les composantes permettent de discriminer les différents modèles M_i . Ainsi, si l'on considère le système décrit par l'équation différentielle :

$$\dot{x} = x \cos(t) + u, \quad (3.19)$$

on peut envisager trois modèles de ce système sur les domaines de validité D_i (on considère dans cet exemple que les modèles sont tels que $\neg D_i = \emptyset$) :

$$\begin{aligned} M_1 : \dot{x} &= x + u \text{ avec } D_1 = \{t = 0\}, \\ M_2 : \dot{x} &= u \text{ avec } D_2 = \left\{t = \frac{\pi}{2}\right\}, \\ M_3 : \dot{x} &= -x + u \text{ avec } D_3 = \{t = \pi\}. \end{aligned} \quad (3.20)$$

Ici, l'état représentatif du système est le temps t (modulo π). Par conséquent, l'espace des modèles est défini par :

$$\begin{aligned} x_r &\in E_M, \\ E_M &= [t = 0 ; t = \pi]. \end{aligned} \quad (3.21)$$

On donne sur la figure suivante une représentation de l'espace des modèles dans lequel sont calculées les distances d_i .

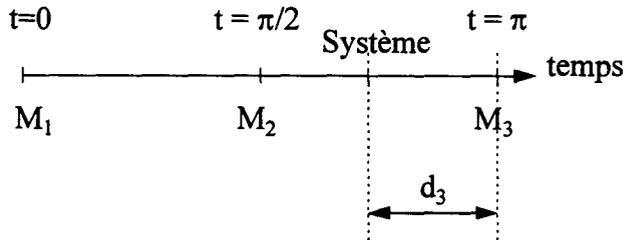


Figure 3.10 : Espace des modèles pour le système (3.19).

Une fois les distances de l'état représentatif du système aux différents domaines calculées, on détermine les confiances dans les modèles M_i de la façon suivante :

$$t_i = \left(1 - \frac{d_i}{\sum_{j=1}^m d_j} \right) \frac{-d_i}{-d_i + d_i}, \quad (3.22)$$

La formulation de cette confiance t_i assure la condition :

$$\text{Plus le modèle } M_i \text{ est loin de } x_r \text{, plus la confiance } t_i \text{ est faible.} \quad (3.23)$$

Néanmoins, lorsque l'état représentatif du système se trouve dans un domaine de validité D_i , la confiance dans les autres modèles M_j , $j \neq i$, est non nulle. Ce phénomène est identique à celui décrit dans la remarque faite dans le premier chapitre pour l'agrégation des règles floues. Afin d'éviter ce problème, une étape de renforcement est ajoutée dans le calcul des confiances t_i . La confiance dans le modèle M_i devient alors :

$$t_i = (1 - d_i') \left(\frac{-d_i}{-d_i + d_i'} \right) \prod_{\substack{j=1 \\ i \neq j}}^m \left(1 - e^{-\left(\frac{d_j'}{\sigma}\right)^2} \right), \quad (3.24)$$

avec :

$$d_i' = \frac{d_i}{\sum_{j=1}^m d_j}. \quad (3.25)$$

Ainsi, si l'état représentatif du système se trouve dans un domaine D_i , les confiances dans tous les autres domaines D_j , $j \neq i$, sont nulles. On peut toutefois remarquer que la confiance dans le modèle M_i n'est pas totale ($t_i < 1$). Néanmoins, comme l'action U_i est la seule envisageable, après l'étape de défuzzification, la commande du système correspondra à celle-ci.

Influence du paramètre σ :

Le paramètre σ qui apparaît dans le terme de renforcement lors du calcul des confiances t_i permet de contrôler la transition entre les différents modèles ou plus exactement la transition lors de la sortie d'un domaine de validité d'un modèle. La figure suivante présente l'allure du terme correctif pour différentes valeurs de σ .

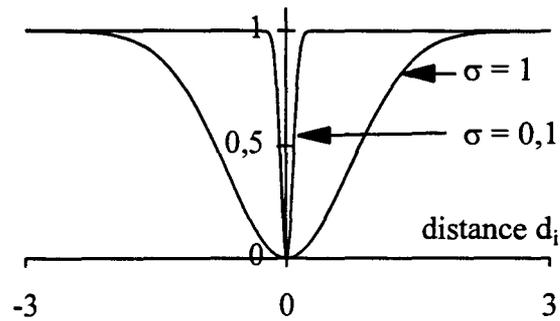


Figure 3.11 : Présentation du terme correctif pour $\sigma = 1$ et $\sigma = 0,1$.

Ainsi, plus σ est important plus les transitions seront brusques et plus la confiance dans un modèle tend à se réduire au domaine de validité de celui-ci.

Exemple :

Considérons le système défini dans l'équation (3.19), nous présentons sur les deux prochaines figures l'évolution des confiances dans les modèles M_i pour différentes valeurs du paramètre σ et pour t variant entre 0 et π secondes.

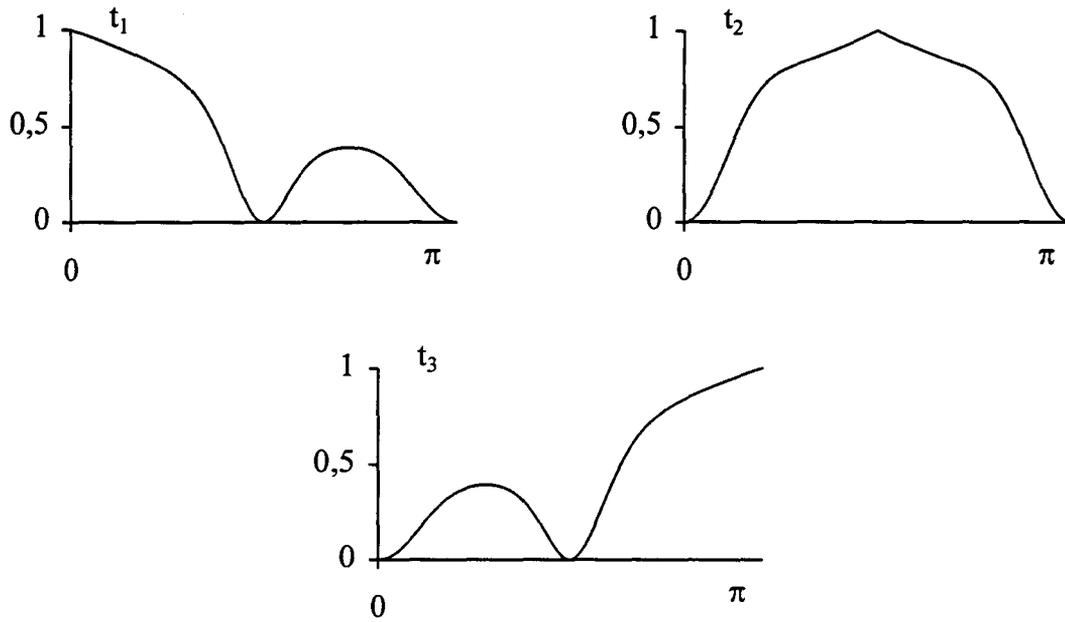


Figure 3.12 : Confiances t_i dans les modèles M_i pour $\sigma = 0,1$.

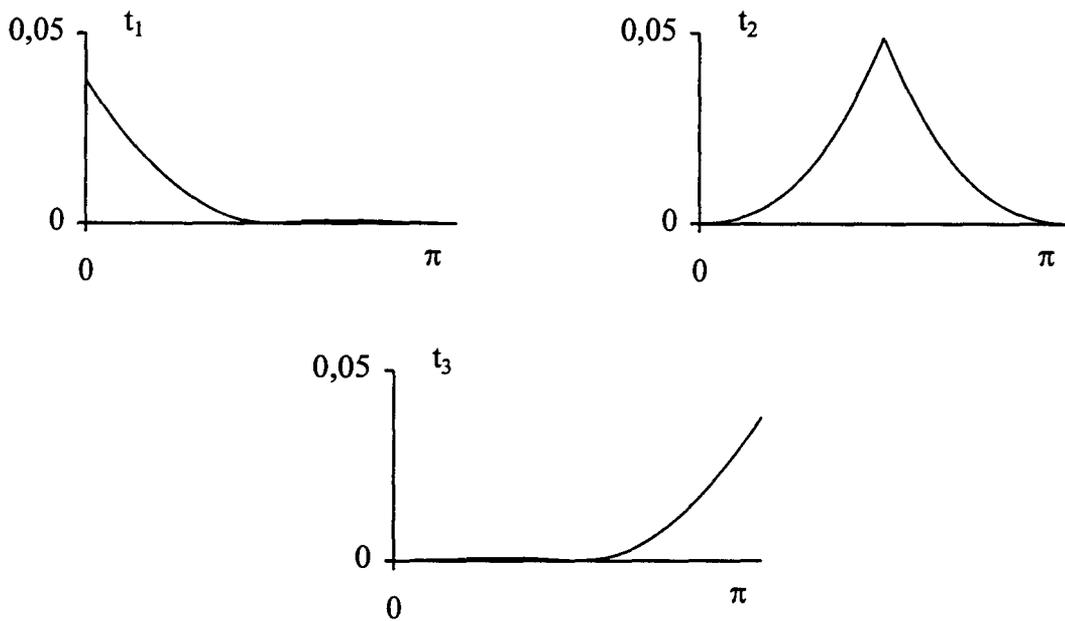


Figure 3.13 : Confiances t_i dans les modèles M_i pour $\sigma = 1$.

Fusion des commandes U_i :

Pour chaque modèle M_i considéré, $i = 1$ à m , on a défini une confiance t_i et une commande U_i . Cette commande est caractérisée par une distribution de possibilité π_i sur l'espace des commandes.

Une distribution de possibilité d'un sous-ensemble flou F est l'équivalent pour la théorie des possibilités de la fonction d'appartenance μ_F présentée dans le chapitre sur la logique floue. Ainsi, une distribution $\pi_F(x)$ est définie par :

$$\begin{aligned} \pi_F : X &\rightarrow [0 ; 1] \\ x &\mapsto \pi_F(x). \end{aligned} \quad (3.26)$$

La différence entre distribution de possibilité et fonction d'appartenance réside dans l'interprétation de la valeur de ces fonctions pour un élément particulier x de X . Pour la logique floue, $\mu_F(x)$ représente le degré d'appartenance de x au sous-ensemble F . A l'inverse, dans la théorie des possibilités, $\pi_F(x)$ est une mesure de la possibilité que x soit la valeur associée à F . Ainsi, si l'on considère le sous-ensemble flou *Jeune*, en logique floue on tente, par l'intermédiaire de la fonction d'appartenance μ_{Jeune} , de classifier, d'identifier dans la base des événements connus, un fait observé, par exemple *Jean a 17 ans*. Avec l'approche possibiliste, à partir du prédicat *Jean est Jeune*, on essaie de déterminer l'âge de Jean.

Dans notre cas, on désire trouver une commande à partir d'un sous-ensemble flou représentant l'action proposée par un modèle. Il s'agit donc bien d'une approche possibiliste. On considère alors les m commandes U_i comme autant de sources d'informations. L'objectif est de déterminer le sous-ensemble flou U_S , caractérisé par la distribution de possibilité π_{u_S} , représentant la commande la plus vraisemblable au regard des différentes propositions (i.e. les distributions de possibilité π_i).

$$\begin{aligned} \text{Modèle 1 : } u_S \in U_1 &= \{u, \pi_1\} \subset U, \dots, \\ \text{Modèle } m : u_S \in U_m &= \{u, \pi_m\} \subset U, \\ \text{Que peut-on en déduire pour } U_S &? \end{aligned} \quad (3.27)$$

Le traitement de l'ensemble des informations conduit au calcul de π_{u_S} , distribution de possibilité globale sur l'espace des commandes. Les méthodes de traitement habituelles sont définies ci-dessous.

Tout d'abord, si toutes les sources d'informations sont cohérentes c'est à dire si elles sont toutes fiables, on peut utiliser un opérateur d'intersection \wedge comme suit :

$$\pi_{u_S} = \bigcap_{i=1}^m \pi_i = \pi_1 \wedge \dots \wedge \pi_m. \quad (3.28)$$

Si par contre, elles sont contradictoires, c'est-à-dire si l'une au moins des sources n'est pas fiable, sans pour autant pouvoir déterminer laquelle, l'opérateur d'union \vee s'impose :

$$\pi_{u_s} = \bigcup_{i=1}^m \pi_i = \pi_1 \vee \dots \vee \pi_m. \quad (3.29)$$

Par l'intermédiaire des coefficients de confiance t_i , on possède une mesure de la fiabilité des informations disponibles (i.e. sur les distributions de possibilité π_i). Comment déterminer alors la distribution de possibilité de la commande U_S ? Une méthode de calcul de π_{u_s} consiste à établir une analogie avec la théorie des probabilités. On obtient :

$$\pi_{u_s} = \sum_{i=1}^m \lambda_i \pi_i, \quad (3.30)$$

avec la condition :

$$\sum_{i=1}^m \lambda_i = 1. \quad (3.31)$$

Par exemple, on peut fixer :

$$\begin{aligned} \forall i = 1 \text{ à } m \\ \lambda_i = \frac{t_i}{\sum_{j=1}^m t_j}. \end{aligned} \quad (3.32)$$

Cette approche "statistique" pose néanmoins un problème. Ainsi, considérons deux sources S_1 et S_2 parfaitement fiables (i.e. $t_1 = t_2 = 1$). La mise en oeuvre de la méthode précédente donne :

$$\pi_{u_s} = \frac{\pi_1 + \pi_2}{2}. \quad (3.33)$$

Par conséquent, l'information déduite est une moyenne entre les deux sources d'informations . Elle est, de fait, moins pertinente que celle obtenue par l'intersection des deux sources. Pour l'exemple présenté sur la figure 3.14, l'application de (3.30) amène à considérer

que la commande u_s la plus vraisemblable appartient à un intervalle. L'utilisation de l'opérateur d'intersection (ici la fonction min) fournit une solution unique.

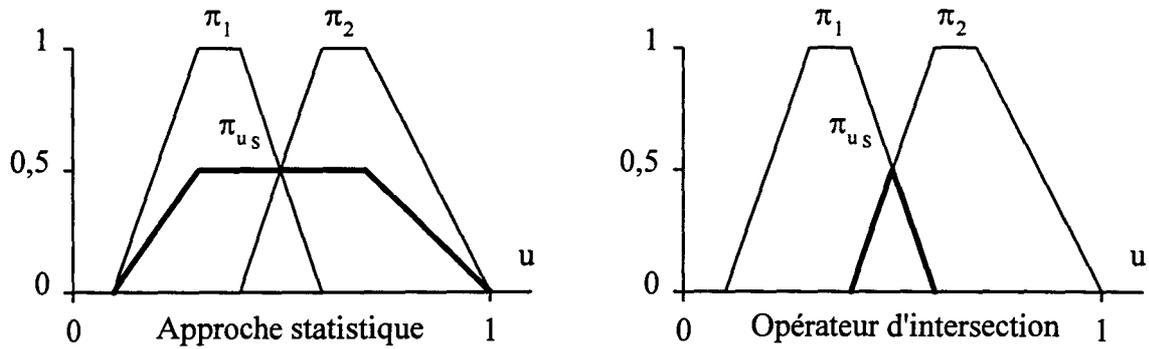


Figure 3.14 : Comparaison pour deux sources fiables.

Pour conserver un sens au calcul de π_{u_s} et fournir l'information la plus judicieuse avant l'étape de défuzzification, il est nécessaire de construire un opérateur permettant de passer, de manière continue, de l'intersection à l'union des $t_i \pi_i$ (ceci permet de rejeter les informations erronées pour lesquelles $t_i = 0$) [Del 1995]. La distribution de possibilité de la commande u_s est alors :

$$\pi_{u_s} = \alpha(t_1, \dots, t_m) \bigcup_{i=1}^m t_i \pi_i + \beta(t_1, \dots, t_m) \bigcap_{i=1}^m t_i \pi_i, \quad (3.34)$$

Dans le cas de deux sources d'informations S_1 et S_2 , α et β respectent les spécifications présentées dans le tableau suivant.

S ₂ \ S ₁	Erronée	Fiable
Erronée	$\alpha = 0$ $\beta = 0$	$\alpha = 1$ $\beta = 0$
Fiable	$\alpha = 1$ $\beta = 0$	$\alpha = 0$ $\beta = 1$

Tableau 3.1 : Valeurs de α et β dans le cas de deux sources d'informations S₁ et S₂.

Ainsi, le terme β est une T-norme tandis que α représente un opérateur OU exclusif (opérateur Xor). Ce dernier est construit comme une extension de l'opérateur booléen défini par :

$$\text{Xor}(x, y) = (x \vee y) \wedge \neg(x \wedge y). \tag{3.35}$$

Pour étendre cette définition, les opérateurs d'union et d'intersection (\vee et \wedge) sont choisis comme des T-normes et des T-conormes. Par exemple, pour les opérateurs duaux *max* et *min* et avec l'opérateur de négation usuel, on obtient :

$$\text{Xor}_{\text{Zadeh}}(x, y) = \min(\max(x, y), 1 - \min(x, y)). \tag{3.36}$$

De même, si l'on choisit les opérateurs probabilistes ($x \wedge y = xy$ et $x \vee y = x+y-xy$), la définition du Xor devient :

$$\text{Xor}_{\text{Prob}}(x, y) = (x + y - xy)(1 - xy). \tag{3.37}$$

Ces deux définitions peuvent être étendues pour un vecteur $t = [t_1, \dots, t_m]^T \subset [0 ; 1]^m$.

$$\text{Xor}_{\text{Zadeh}}(t) = \min\left(\max_{i=1 \text{ à } m}(t_i), 1 - \min_{i=1 \text{ à } m}(t_i)\right), \tag{3.38}$$

$$\text{Xor}_{\text{Prob}}(t) = \left(\sum_{i=1}^m t_i \left(1 - \sum_{j>i}^m t_j \left(1 - \sum_{k>j}^m \dots - t_m\right)\right)\right) \left(1 - \prod_{i=1}^m t_i\right). \tag{3.39}$$

Ainsi, la détermination de la distribution de possibilité π_{u_s} par la fusion des m couples (π_i, t_i) est donnée par :

$$\pi_{u_s} = \text{Xor}(t) \bigcup_{i=1}^m t_i \pi_i + \bigcap_{i=1}^m t_i \bigcap_{i=1}^m t_i \pi_i . \quad (3.40)$$

Calcul de la commande u_s :

La détermination de la commande à appliquer au système est obtenue par la mise en oeuvre d'un opérateur de défuzzification. Ainsi la commande u_s est :

$$u_s = D_f(\{u \in V, \pi_{u_s}\}) . \quad (3.41)$$

Par exemple, si l'on considère que les commandes U_i sont des singletons indépendants les uns des autres (i.e. $U_i \equiv U_j$ si et seulement si $i = j$) et en utilisant une défuzzification barycentrique on obtient la commande :

$$u_s = \frac{\sum_{i=1}^m U_i \pi_{u_s}(U_i)}{\sum_{i=1}^m \pi_{u_s}(U_i)} . \quad (3.42)$$

5 Exemples d'application utilisant des modèles locaux

Afin de montrer les apports d'une commande multi-modèle, nous allons présenter dans cette partie deux applications différentes. La première est la commande d'un système SISO non-linéaire par un ensemble de commandes par mode glissant [Dub 1994]. Ensuite, nous présenterons une commande sous-optimale multi-critère obtenue avec l'approche multi-modèle [Dub 1995b].

5.1 Commande multi-modèle par mode glissant

5.1.1 Commande par mode glissant

L'idée de la commande par mode glissant [Per 1994] est de réaliser une commande modifiant de manière discontinue l'expression du système commandé. On obtient alors un système à structure variable ([Utk 1965], [Utk 1992]) dans lequel l'hypersurface $S(x) = 0$ délimite les différents systèmes et vers laquelle converge ceux-ci.

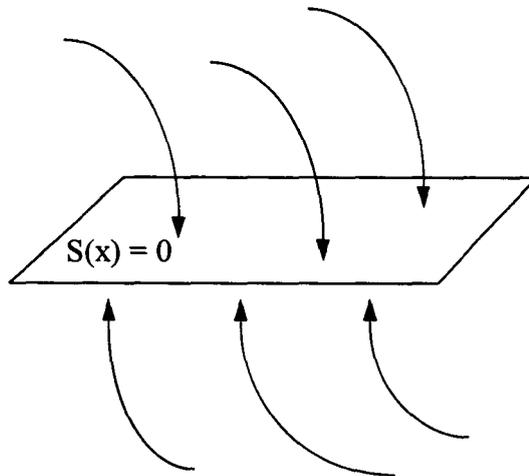


Figure 3.14 : Principe de la commande par mode glissant.

Considérons le système SISO décrit par l'équation différentielle :

$$y^{(n)} = f(x) + g(x)u, \quad (3.43)$$

où $x = [y, \dots, y^{(n-1)}]^T \in \mathfrak{R}^n$ est le vecteur d'état du système et $u \in \mathfrak{R}$ est l'action de commande.

Le modèle du système est de la forme :

$$y^{(n)} = \tilde{f}(x) + \tilde{g}(x)u, \quad (3.44)$$

où $\tilde{f}(x)$ et $\tilde{g}(x)$ sont les estimations des fonctions $f(x)$ et $g(x)$. Les erreurs de modélisation sont alors :

$$\begin{aligned} \Delta f &= f - \tilde{f}, \\ \Delta g &= g - \tilde{g}. \end{aligned} \quad (3.45)$$

La surface de glissement S est décrite par l'équation :

$$S = \sum_{i=0}^{n-1} a_{i+1} e^{(i)}, \quad (3.46)$$

où les a_i , $i = 1$ à n et avec $a_n = 1$, sont les coefficients d'un polynôme de Hurwitz, $e = y - y_d$ est l'erreur de poursuite entre la sortie du système et son évolution désirée et $e^{(i)}$ est la i -ème dérivée par rapport au temps de cette erreur. On désire assurer la condition d'attractivité de l'hypersurface $S = 0$, c'est-à-dire :

$$\text{si } S \neq 0, \dot{S} < 0. \quad (3.47)$$

Le calcul de la dérivée de S donne :

$$\begin{aligned} \dot{S} &= \sum_{i=1}^{n-1} a_i e^{(i)} + e^{(n)} \\ &= \sum_{i=1}^{n-1} a_i e^{(i)} - y_d^{(n)} + \tilde{f} + \tilde{g}u + (\Delta f + \Delta g u). \end{aligned} \quad (3.48)$$

Alors la commande (3.49) assure l'attractivité de l'hypersurface $S = 0$ et maintient l'état sur celle-ci une fois atteinte.

$$u(t) = \hat{u}(t) - \frac{k_{SMC}}{\tilde{g}} \text{sign}(S) \quad (3.49)$$

Dans cette équation,

$$\hat{u}(t) = \frac{1}{\tilde{g}} \left(-pS - \tilde{f} + y_d^{(n)} - \xi \right), \quad (3.50)$$

où p est une constante positive et $\xi = \sum_{i=1}^{n-1} a_i e^{(i)}$, est la commande qui assure pour un système connu la condition :

$$\dot{S} = -pS. \quad (3.51)$$

Pour le système réel la dynamique de S est :

$$\dot{S} = -pS - k_{SMC} \left(1 + \frac{\Delta g}{\tilde{g}} \right) \text{sign}(S) + \Delta f + \Delta g \hat{u}. \quad (3.52)$$

Par conséquent, afin de conserver la vitesse de convergence de l'état vers $S = 0$ le gain k_{SMC} doit être tel que :

$$k_{SMC} \geq \left| \frac{\Delta f + \Delta g \hat{u}}{1 + \frac{\Delta g}{\tilde{g}}} \right|_{\text{pire des cas}}. \quad (3.53)$$

Afin d'éviter les phénomènes haute-fréquence sur la commande lorsque l'erreur se rapproche de la surface de glissement, Slotine et Sastry dans [Sol 1983] ont proposé de remplacer la fonction $\text{sign}(x)$ par la fonction saturation $\text{sat}(x)$ présentée ci-dessous.

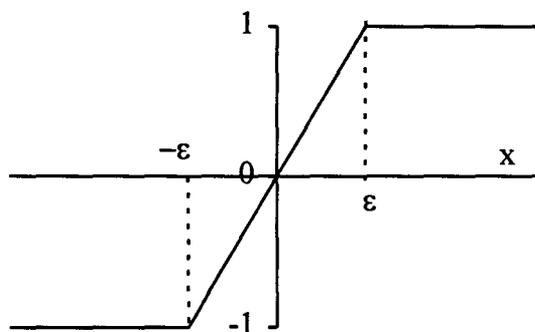


Figure 3.15 : Fonction saturation.

Pour accroître les performances de la commande réalisée, la zone linéaire de taille 2ϵ de la fonction $\text{sat}(x)$ doit être aussi petite que possible. Pourtant, cette zone est inversement proportionnelle au gain k_{SMC} , qui augmente avec les perturbations sur le système (erreurs de modélisation, bruits, ...). Par conséquent, un compromis doit être fait entre la robustesse et les

performances de la commande par mode glissant. L'objectif de la commande multi-modèle utilisant des commandes par mode glissant est de parvenir à accroître les performances de poursuite tout en évitant les commutations haute-fréquence sur la commande.

5.1.2 Commande multi-modèle

Considérons m modèles M_i de la forme :

$$y^{(n)} = \tilde{f}_i(x) + \tilde{g}_i(x)u, \quad (3.54)$$

pour lesquels les erreurs sont

$$\begin{aligned} \Delta f_i &= f - \tilde{f}_i, \\ \Delta g_i &= g - \tilde{g}_i, \end{aligned} \quad (3.55)$$

et auxquels sont associés les commandes u_i

$$u_i(t) = \hat{u}_i(t) - \frac{k_i}{\tilde{g}_i} \text{sign}(S), \quad (3.56)$$

avec :

$$\hat{u}_i(t) = \frac{1}{\tilde{g}_i} (-pS - \tilde{f}_i + y_d^{(n)} - \xi). \quad (3.57)$$

La commande globale appliquée au système est :

$$u_s = \frac{\sum_{i=1}^m u_i \pi_{u_s}(u_i)}{\sum_{i=1}^m \pi_{u_s}(u_i)}. \quad (3.58)$$

La dynamique de S est alors :

$$\dot{S} = \xi - y_d^{(n)} + (f + gu_s). \quad (3.59)$$

où l'on peut écrire $f(x)$ sous la forme :

$$f(x) = \frac{\sum_{i=1}^m f(x)\pi_{u_s}(u_i)}{\sum_{i=1}^m \pi_{u_s}(u_i)} = \frac{\sum_{i=1}^m (\tilde{f}_i(x) + \Delta f_i)\pi_{u_s}(u_i)}{\sum_{i=1}^m \pi_{u_s}(u_i)}. \quad (3.60)$$

En remplaçant la fonction $f(x)$ par cette expression, la dynamique de S devient :

$$\dot{S} = \xi - y_d^{(n)} + \frac{\sum_{i=1}^m (f_i + \Delta f_i + g u_i)\pi_{u_s}(u_i)}{\sum_{i=1}^m \pi_{u_s}(u_i)}, \quad (3.61)$$

comme u_i est une commande par mode glissant, on obtient

$$\dot{S} = -pS + \frac{\sum_{i=1}^m \left(-k_i \left(1 + \frac{\Delta g_i}{\tilde{g}_i} \right) \text{sign}(S) + \Delta f_i + \Delta g_i \hat{u}_i \right) \pi_{u_s}(u_i)}{\sum_{i=1}^m \pi_{u_s}(u_i)}. \quad (3.62)$$

Afin d'assurer la vitesse minimale de convergence vers la surface de glissement, la condition suivante doit être vérifiée.

$$S \sum_{i=1}^m \left(-k_i \left(1 + \frac{\Delta g_i}{\tilde{g}_i} \right) \text{sign}(S) + \Delta f_i + \Delta g_i \hat{u}_i \right) \pi_{u_s}(u_i) \leq 0 \quad (3.63)$$

Ainsi, le terme correctif apporté par la distribution de possibilité π_{u_s} permet une compensation mutuelle des perturbations sur les modèles M_i .

5.1.3 Simulations

Nous présentons ici les résultats de simulations de la commande d'un pendule inversé par les deux méthodes présentées ci-dessus. Le système est identique à celui présenté dans le premier chapitre. Il obéit à l'équation différentielle :

$$\ddot{\theta} = \frac{g \sin \theta - \cos \theta \left(\frac{F + m l \dot{\theta}^2 \sin \theta}{m + m_c} \right)}{l \left(\frac{4}{3} - \frac{m \cos^2 \theta}{m + m_c} \right)}, \quad (3.64)$$

où m et m_c sont respectivement les masses du bras et du chariot, l la demi-longueur du bras, F la force appliquée au chariot. Un modèle grossier de ce système autour d'un angle θ_i (point de fonctionnement du modèle M_i) peut être pris de la forme :

$$\ddot{\theta} = \frac{g \sin(\theta_i) - \frac{F}{m + m_c} \cos(\theta_i)}{l \left(\frac{4}{3} - \frac{m}{m + m_c} \cos(\theta_i) \right)}, \quad (3.65)$$

On désire suivre la trajectoire suivante

$$y_d = \frac{1}{2} \theta_{(t=0)} (1 + \cos(\pi t)), \quad (3.66)$$

dont la figure ci-dessous présente l'évolution pour un angle initial égal à 10° .

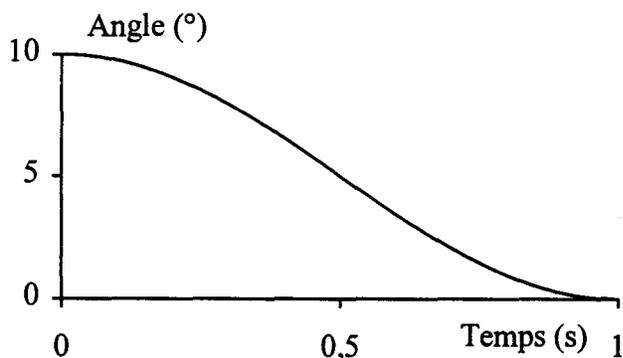


Figure 3.16 : Sortie désirée y_d .

La commande par mode glissant conventionnelle est définie pour le modèle au point d'équilibre (i.e. $\theta = 0$). Pour la commande multi-modèle, les domaines de validité sont $D_i = \{\theta_i\}$ et on étudiera l'influence des domaines de non-validité $\neg D_i$.

Modèle	θ_i	Cas 1 : $-D_i$	Cas 2 : $-D_i$
M_1	-10	\emptyset	\mathcal{R}^+
M_2	0	\emptyset	\emptyset
M_3	10	\emptyset	\mathcal{R}^-

Tableau 3.2 : Choix dans la commande multi-modèle.

La surface de glissement est caractérisée par l'équation

$$S = 20e + \dot{e} . \tag{3.67}$$

Les figures ci-dessous présentent l'évolution de l'erreur de poursuite et de la commande pour la commande par mode glissant conventionnelle avec $k_{SMC} = 4,5$ et $\epsilon_{SMC} = 0,015$.

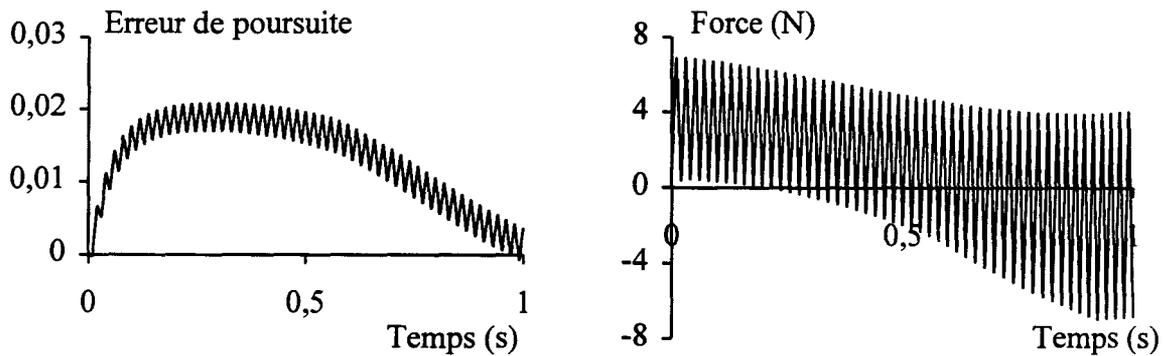


Figure 3.17 : Commande par mode glissant.

Ensuite, nous avons testé la commande multi-modèle pour laquelle aucun domaine de non-validité n'a été défini. De plus, la commande par mode glissant pour le modèle M_2 est identiques à la commande précédente (i.e. $k_2 = k_{SMC}$ et $\epsilon_2 = \epsilon_{SMC}$).

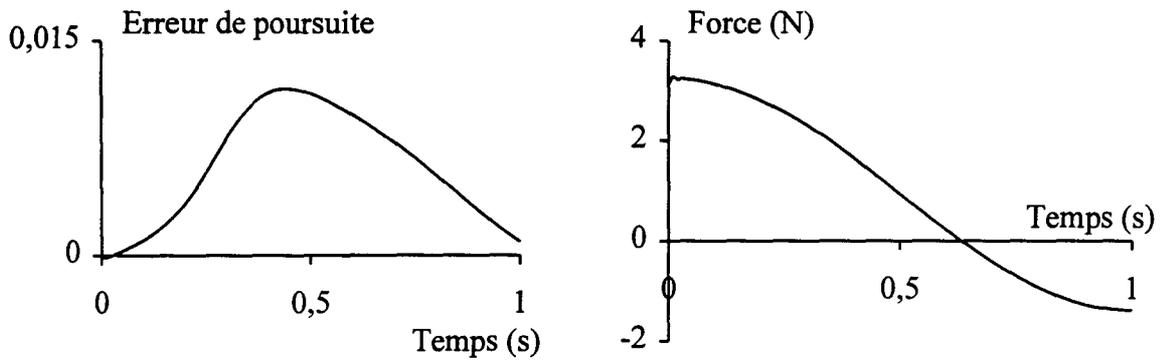


Figure 3.18 : Commande multi-modèle sans domaine de non-validité.

On observe que l'utilisation d'une approche multi-modèle permet de supprimer les commutations haute-fréquence sur la commande sans diminuer les performances de poursuite de la commande. La figure suivante présente l'évolution des confiances t_i dans les modèles M_i .

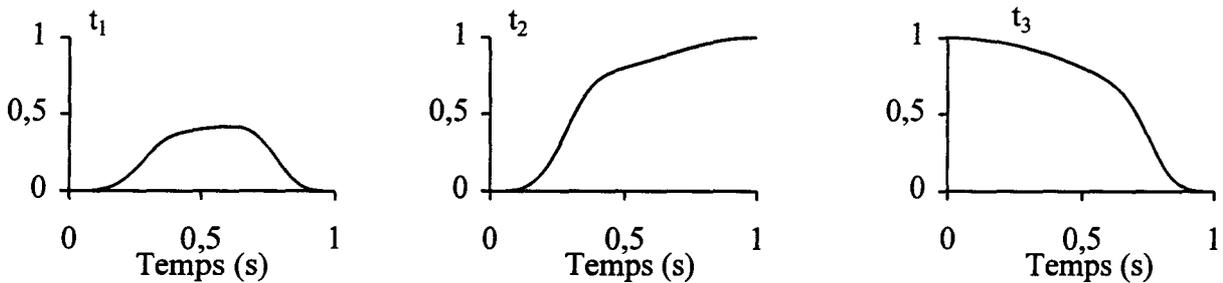


Figure 3.19 : Evolution des confiances t_i dans les modèles M_i .

Afin de limiter l'influence des erreurs de modélisation sur les modèles extrêmes M_1 et M_3 , on définit comme domaines de non-validité $\neg D_1 = \mathcal{R}^+$ et $\neg D_3 = \mathcal{R}^-$.

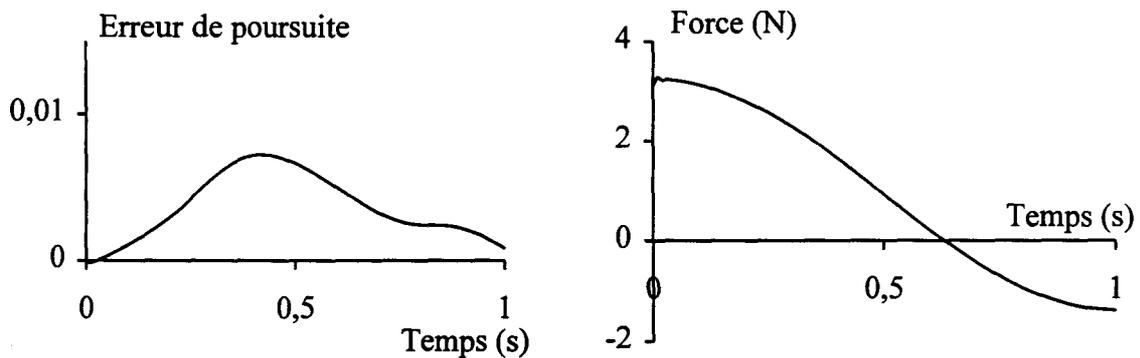
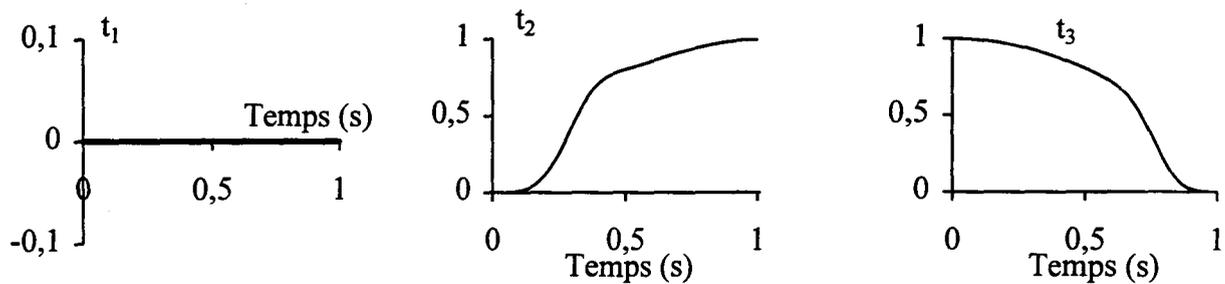


Figure 3.20 : Commande multi-modèle où sont définis des domaines de non-validité $\neg D_1 = \mathcal{R}^+$ et $\neg D_3 = \mathcal{R}^-$.

La figure qui suit présente l'évolution des confiances t_i dans les trois modèles M_i .

Figure 3.21 : Evolution des confiances t_i dans les modèles M_i .

Les résultats des simulations confirment la cohérence des confiances dans les modèles définis et la souplesse d'utilisation de la commande multi-modèle. En effet, elle permet "d'optimiser" les couples (k_i, ε_i) de chaque commande par mode glissant. De plus, avec l'utilisation de domaines de non-validité, il est possible de limiter leurs emplois pour des domaines restreints par l'utilisateur. Ainsi, la commande multi-modèle contourne le problème de la diminution des performances d'une unique commande robuste par la réalisation un ensemble de commandes, robustes localement, et dont on assure la robustesse globale par une transition continue d'une commande à une autre.

5.2 Commande sous-optimale multi-critère

5.2.1 Position du problème

Pour des problèmes de résolution d'équations différentielles, la création d'une commande optimale est limitée dans le nombre de critères à prendre en compte. Ainsi, bien qu'il soit possible de définir les commandes optimales U_i pour m critères J_i , le calcul de la commande U pour le critère $J = \sum J_i$ est le plus souvent impossible à réaliser. Néanmoins, une connaissance minimale du système permet discerner le domaine D_i où la commande liée à un critère particulier J_i doit être active pour satisfaire le critère global J . De ce fait l'approche multi-modèle pour la création d'une commande sous-optimale multi-critère semble adéquate.

5.2.2 Commande sous-optimale d'un système du premier ordre

Nous présentons ici l'utilisation d'une commande multi-modèle pour un système du premier ordre pour lequel on désire atteindre une consigne en minimisant l'erreur et l'énergie dépensée. Ce système a été choisi afin de pouvoir comparer les résultats de notre approche avec ceux de la commande optimale. Le système est défini par l'équation différentielle

$$\tau \dot{x} = ku - x, \quad (3.68)$$

où k et τ sont respectivement le gain statique et la constante de temps du système. On désire qu'à l'instant t_f la consigne x_f soit atteinte (à $t = t_0$ $x = x_0$). Le critère énergétique est

$$J_1 = \int_{t_0}^{t_f} \frac{1}{2} u^2 dt. \quad (3.69)$$

La commande optimale ([BOR 1990]) en boucle fermée correspondante est

$$U_1(t) = \frac{2e^{-(t_f-t)/\tau} \left(x_f - x e^{-(t_f-t)/\tau} \right)}{k \left(1 - e^{-2(t_f-t)/\tau} \right)}. \quad (3.70)$$

Le deuxième critère a pour objectif de minimiser l'erreur :

$$J_2 = \int_{t_0}^{t_f} \frac{1}{2} (x - x_f)^2 dt, \quad (3.71)$$

avec $|u| \leq U_{\max}$.

On assure ce critère par une commande de type bang-bang :

$$\begin{aligned} U_2(t) &= U_{\max} \text{sign}(x_f - x), \\ \text{sign}(x) &= 1 \text{ if } x > 0, \\ &= 0 \text{ if } x = 0, \\ &= -1 \text{ if } x < 0. \end{aligned} \quad (3.72)$$

Le critère global J est

$$J = \int_{t_0}^{t_f} \frac{1}{2} \left(k_1 u^2 + k_2 (x - x_f)^2 \right) dt, \quad (3.73)$$

où les constantes positives k_i permettent de modifier l'importance relative des deux critères J_1 et J_2 . La commande optimale est alors

$$U_{\text{opt}}(t) = \gamma x + \beta, \quad (3.74)$$

avec les expressions de β et γ suivantes :

$$\beta = x_f \frac{2\alpha + rk^2(\alpha^2 + 1)}{k(\alpha^2 - 1)\sqrt{1 + rk^2}},$$

$$\gamma = \frac{1}{k} \left(1 - \frac{\alpha^2 + 1}{\alpha^2 - 1} \sqrt{1 + rk^2} \right),$$
(3.75)

dans lesquelles

$$\alpha = e^{\frac{\sqrt{1+rk^2}}{\tau}(t_f-t)},$$

$$r = \frac{k_2}{k_1}.$$
(3.76)

Analyse de l'expert :

Afin de minimiser l'erreur, le domaine de validité de la commande U_2 correspond aux points éloignés de la consigne. Par contre, afin de réduire l'énergie de commande et accroître la robustesse, la commande U_1 doit être active lorsque l'on se rapproche de la consigne.

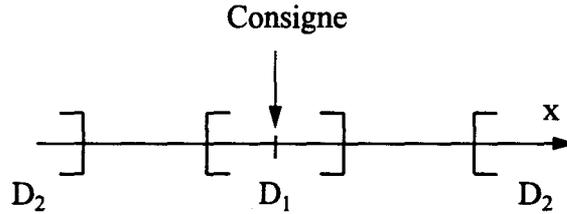


Figure 3.22 : Définition des domaines de validité D_i .

La taille de ces domaines dépend de l'importance accordée aux différents critères élémentaires J_i (i.e. les coefficients k_i). Le tableau suivant donne les domaines de validité D_i pour les cas extrêmes.

k_1	k_2	D_1	D_2
0	1	$\{\emptyset\}$	\mathfrak{R}
1	0	\mathfrak{R}	$\{\emptyset\}$

Tableau 3.3 : Domaines de validité pour les cas extrêmes.

5.2.3 Simulations

Les simulations suivantes ont été réalisées pour les coefficients $k_1 = 1$ et $k_2 = 10$ afin de donner une importance équivalente aux deux critères. La valeur initiale de x est $x_{(t=0)} = 0$ et la consigne est $x = 1$. De ce fait, les domaines de validité choisis sont :

$$D_1 = \{0\} ; D_2 = \{1\} \tag{3.77}$$

Pour la commande optimale U_{opt} , les évolutions de la commande et de la sortie sont présentées sur la figure ci-dessous. Le coût minimal J_{opt} est alors de 4,7.

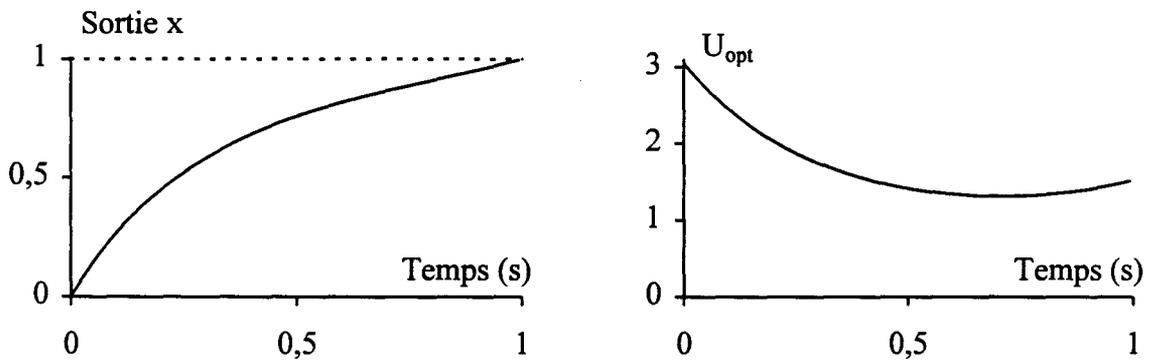


Figure 3.23 : Simulation pour la commande optimale.

Les simulations suivantes représentent les évolutions de la sortie et de la commande lorsqu'une seule commande U_i est appliquée au système. Dans ce cas, les coûts respectifs sont $J_{U_1} = 6$ et $J_{U_2} = 5,9$.

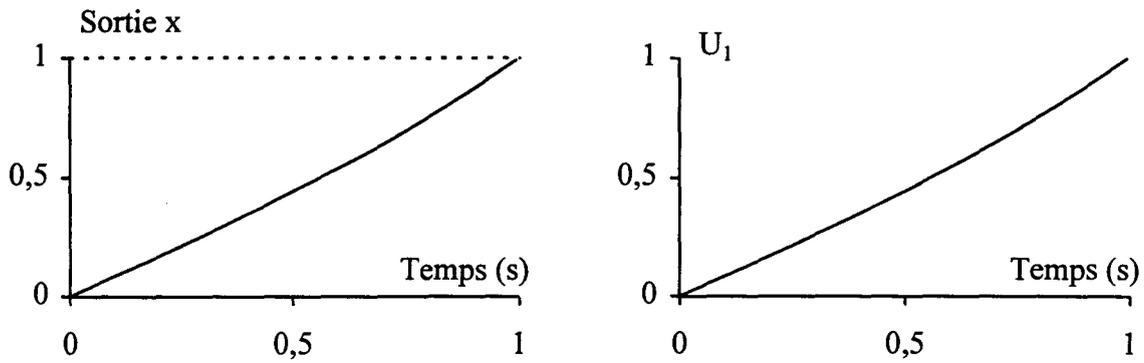


Figure 3.24 : Simulation pour la commande U_1 .

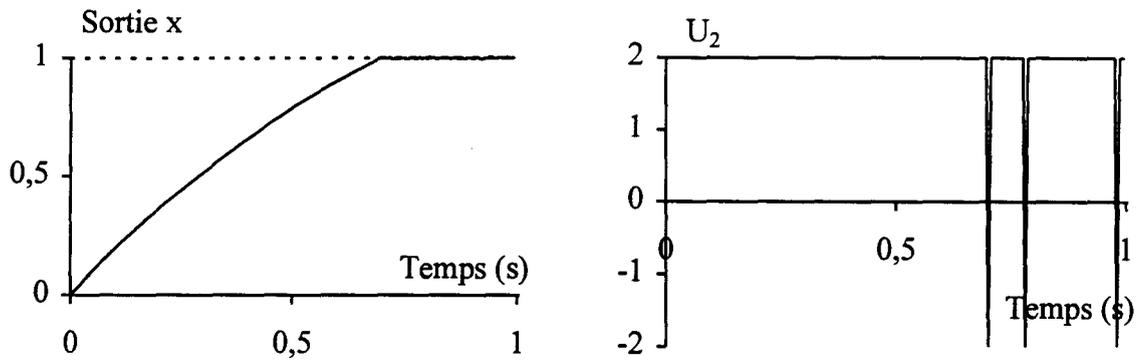


Figure 3.25 : Simulation pour la commande U_2 .

Enfin, la commande multi-modèle u_s à un coût J_{u_s} de 4,9 très proche du coût optimal.

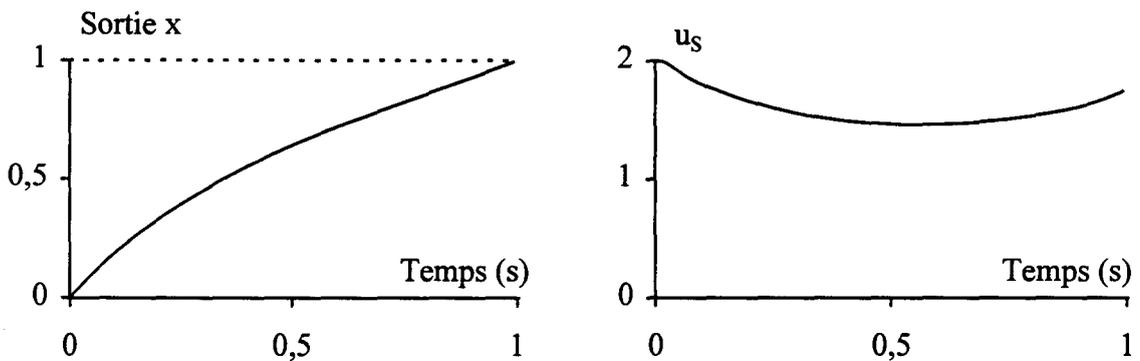


Figure 3.26 : Simulation pour la commande multi-modèle u_s .

Ainsi, l'approche multi-modèle développée ici pour la création d'une commande sous-optimale multi-critère semble prometteuse. En effet, elle génère, une fois les domaines d'activité des commandes optimales élémentaires définis, une commande performante avec la possibilité d'adapter la taille des domaines de validité des modèles afin d'accroître les performances.

6 Utilisation de modèles génériques

6.1 Introduction

A partir d'un système réel, il est possible d'obtenir des modèles dont on ne peut pas définir avec suffisamment de certitude le domaine de validité pour utiliser la commande développée dans la partie précédente. C'est le cas, par exemple, d'un système économique comme le système boursier. En effet, l'évolution du cours d'une action est liée à des phénomènes non mesurables, tels que des événements politiques, économiques ou sociaux, ou des aspects psychologiques comme la confiance d'un grand nombre de personnes dans un modèle particulier ou l'aspect moral d'une action (favoriser une grève, délit d'initiés, ...).

Informations extérieures non mesurables

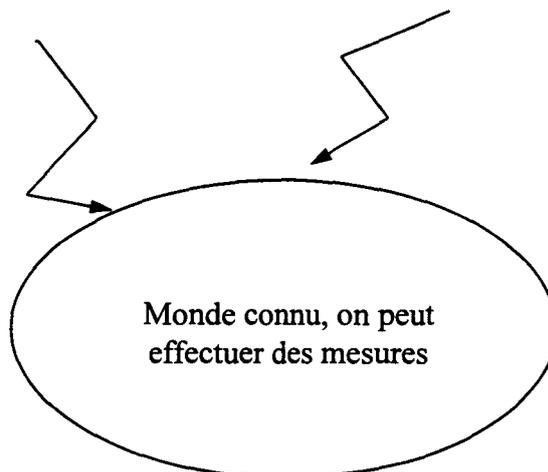


Figure 3.27 : Système candidat à l'utilisation de modèles génériques.

Néanmoins, l'existence et l'utilisation d'un modèle implique que, pour au moins une configuration du système global, l'erreur de modélisation est faible. On conçoit alors aisément l'intérêt d'utiliser plusieurs de ces modèles afin d'identifier le système global. La confiance à accorder à un modèle dépend de la justesse de la prédiction réalisée par ce dernier. On peut prendre pour modèle du système le modèle donnant la prédiction la plus précise. Notre approche consiste à utiliser la logique floue afin d'extraire une solution satisfaisante à partir de l'ensemble des données disponibles.

6.2 Commande floue multi-modèle utilisant des modèles génériques

6.2.1 Structure

La structure de la commande multi-modèle est la suivante :

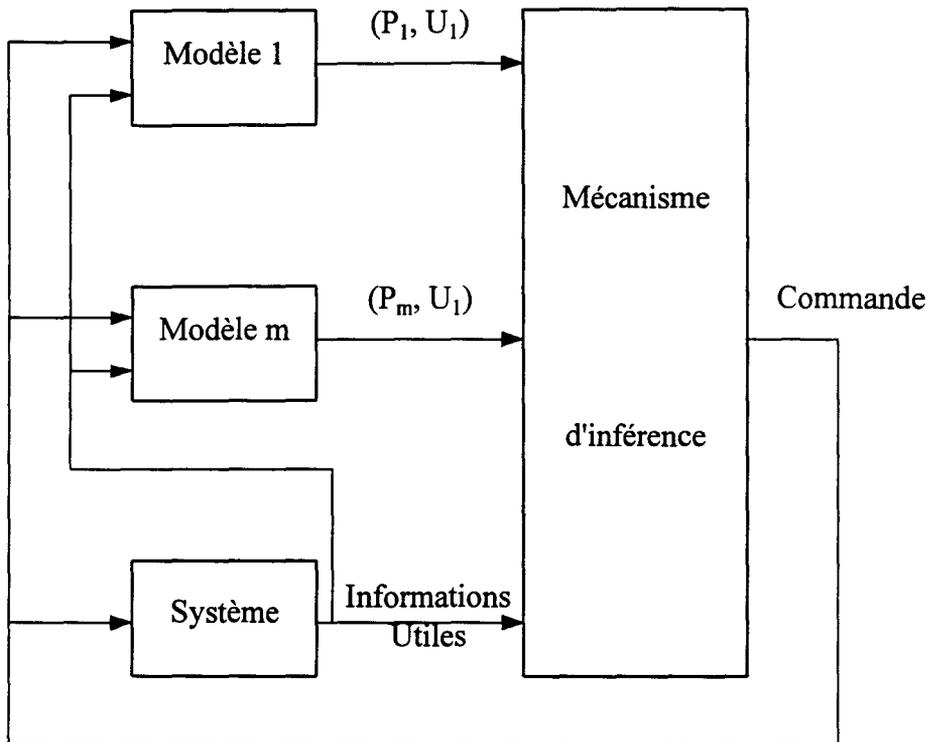


Figure 3.28 : Structure de la commande multi-modèle par utilisation de modèles génériques.

Elle se compose de m modèles M_i qui, à partir de la commande et des informations disponibles sur le système, fournissent des prédictions P_i sur l'évolution futur de ce dernier et proposent des commandes U_i . Le mécanisme d'inférence détermine alors les confiances dans les différents modèles et calcule la commande à appliquer au système.

6.2.2 Principe de fonctionnement

Le mécanisme d'inférence utilise des règles de la forme :

Plus la prédiction est juste plus la confiance dans le modèle doit être importante. (3.43)

La détermination de la commande u_s suit les mêmes étapes que dans le cas de l'utilisation de modèles locaux.

- 1- Calcul des confiances dans les modèles M_i ,
- 2- Calcul du sous-ensemble flou U_S par la fusion des commandes U_i proposées,
- 3- Défuzzification de U_S .

Calcul des confiances

Le calcul des confiances dans les modèles M_i est basé sur l'erreur entre les prédictions réalisées par les modèles et l'état du processus. Cette erreur correspond à la distance entre la prédiction et l'état du système

$$e_i = \|\tilde{x}^i - x\|, \quad (3.78)$$

où \tilde{x}^i est la prédiction de l'état x du système par le modèle M_i . On peut choisir par exemple :

$$\|\tilde{x}^i - x\| = \left(\sum_{j=1}^n (\tilde{x}_j^i - x_j)^2 \right)^{\frac{1}{2}}. \quad (3.79)$$

La confiance dans un modèle M_i est décrite par

$$t_i = 1 - \frac{e_i}{\sum_{j=1}^m e_j} \quad (3.80)$$

Afin d'assurer que, dans le cas où un modèle M_i est totalement approprié (i.e. $e_i = 0$) la confiance dans les autres modèles soit nulle, une étape de renforcement est mise en oeuvre. La confiance devient alors :

$$t_i = (1 - e_i') \prod_{\substack{j=1 \\ i \neq j}}^m \left(1 - e^{-\left(\frac{e_j'}{\sigma}\right)^2} \right), \quad (3.81)$$

où σ une constante positive et

$$e_i' = \frac{e_i}{\sum_{j=1}^m e_j}. \quad (3.82)$$

Fusion des commandes U_i

Comme dans le cas de l'utilisation de modèles locaux, la fusion des commandes U_i est réalisée en utilisant la théorie des possibilités. La distribution de possibilité de la commande u_s est :

$$\pi_{u_s} = \text{Xor}(t) \bigcup_{i=1}^m t_i \pi_i + \bigcap_{i=1}^m t_i \bigcap_{i=1}^m t_i \pi_i . \quad (3.83)$$

Calcul de la commande u_s

On utilise un opérateur de défuzzification afin de déterminer la commande u_s à appliquer au système. Par exemple, si les commandes U_i sont des singletons indépendants les uns des autres, la méthode de défuzzification barycentrique donne :

$$u_s = \frac{\sum_{i=1}^m U_i \pi_{u_s}(U_i)}{\sum_{i=1}^m \pi_{u_s}(U_i)} . \quad (3.84)$$

6.3 Exemple d'application utilisant des modèles génériques

6.3.1 Présentation

Pour montrer l'intérêt de l'utilisation de modèles génériques, nous allons utiliser l'approche multi-modèle afin de représenter dans la base des modèles choisis le système du premier ordre non-stationnaire suivant :

$$\tau \dot{x} = ku - x , \quad (3.85)$$

où k et τ sont respectivement le gain statique et la constante de temps du système et dont les évolutions au cours du temps sont données par les équations :

$$\begin{aligned} k &= 10 + 8 \sin(6\pi t) \\ \tau &= 10 - 4 \cos(3\pi t) \end{aligned} \quad (3.86)$$

On donne sur la figure 3.29 l'aspect de ces évolutions.

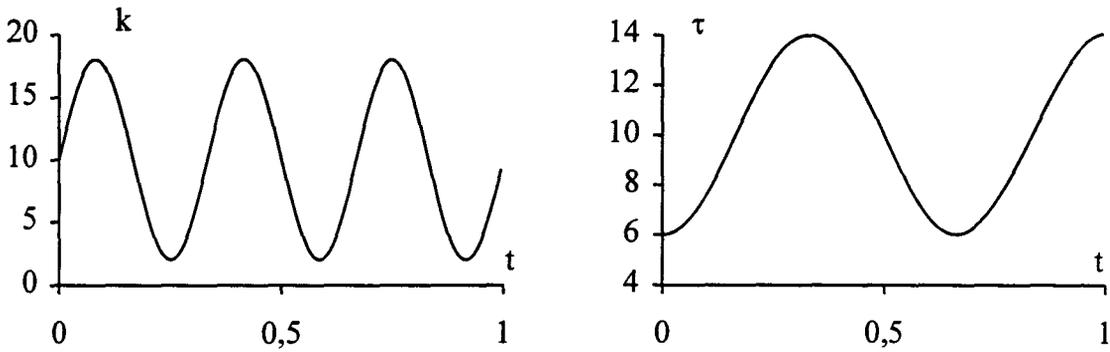


Figure 3.29 : Evolution du gain statique k et de la constante de temps τ du système.

6.3.2 Simulations

On considère trois modèles de ce système correspondant, pour les premier et le troisième modèles, aux variations extrêmes des couples (k, τ) et pour le second aux valeurs moyennes de (k, τ) . Ces modèles sont donc décrits par les équations différentielles :

$$\begin{aligned}
 M_1 : \dot{x} &= -\frac{1}{14}x + \frac{1}{7}u, \\
 M_2 : \dot{x} &= -\frac{1}{7}x + \frac{9}{7}u, \\
 M_3 : \dot{x} &= -\frac{1}{6}x + 3u.
 \end{aligned} \tag{3.87}$$

Le choix de la commande U_i :

$$U_i = \frac{1}{k_i}(\tau_i - p)x, \tag{3.88}$$

où p est une constante positive, assure pour le modèle M_i la dynamique souhaitée

$$\forall i, M_i : \dot{x} = -px. \tag{3.89}$$

Les figures ci-dessous présentent les évolutions des confiances t_i dans les différents modèles M_i et de la sortie du système.

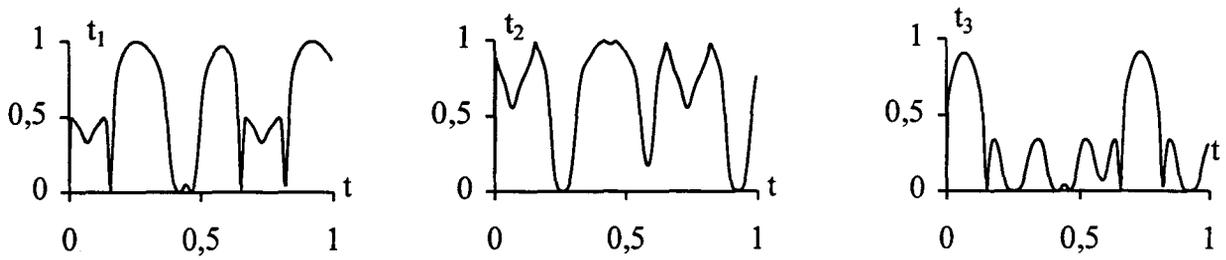


Figure 3.30 : Evolution des confiances t_i dans les modèles M_i .

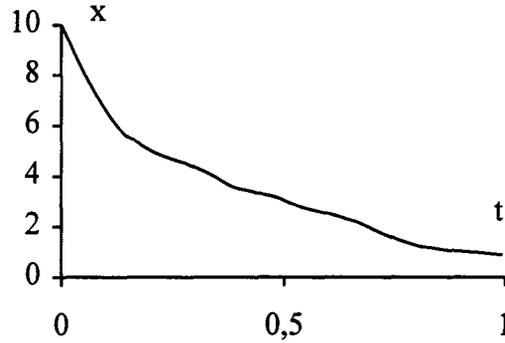


Figure 3.31 : Evolution de la sortie du système.

L'utilisation de modèles génériques permet, par l'intermédiaire des coefficients de confiance t_i , de représenter le système dans la base des modèles M_i . De plus, la commande générée assure la convergence du système vers 0 malgré l'aspect non-stationnaire de ses paramètres.

7 Conclusions

La commande multi-modèle présentée dans ce chapitre offre la possibilité d'intégrer, au niveau des commandes élémentaires, le savoir faire développé par la communauté scientifique (les commandes robustes [Ous 1994] telle que la commande CRONE [Ous 1991], la commande optimale [Bor 1990], la commande par mode glissant [Utk 1992], ...).

Ainsi, par l'approche originale basée sur la fusion (en utilisant une approche logique) des commandes élémentaires, on assure pour la commande d'un système complexe la réalisation, dans des domaines définis, de critères distincts. Entre ces domaines, la commande appliquée est une combinaison de ces commandes élémentaires ce qui permet un passage continu d'une commande à une autre.

L'utilisation de modèles génériques permet la représentation du système dans la base des modèles choisis. Ainsi, il est concevable d'utiliser les différentes confiances dans les modèles pour développer une commande multi-modèle adaptative (afin d'adapter les commandes élémentaires ou les modèles).

Conclusion générale

Position du problème

Depuis la première application industrielle de la théorie de la logique floue en 1970, le nombre d'applications ne cesse de s'accroître et ce tout particulièrement dans la commande des systèmes. Ce phénomène s'explique par la possibilité, lors de construction d'un système flou, d'intégrer l'analyse (e.g. stratégie de commande, identification) qualitative d'un expert. Cette analyse se met sous la forme d'une base de règles du type:

$$R^i : \text{SI } x \text{ est } A_i \text{ ALORS } y \text{ est } B_i,$$

où A_i et B_i sont représentatifs de termes linguistiques tels que *Grand* ou *Petit*.

Deux problèmes se posent néanmoins. Le premier découle de l'aspect heuristique de la création des règles floues. Est-il possible de créer un système flou assurant l'objectif désiré ? Le deuxième est le grand nombre de paramètres fixés de manière arbitraire et qui nécessite une période de réglage avant d'obtenir des résultats convenables.

Une commande floue est-elle farfelue ?

Avec les travaux de Wang [Wan 1992a] et de Kosko [Kos 1992], il a été montré qu'un système à base de règles du type précédent est capable d'approximer avec la précision désirée toute fonction continue f d'un ensemble compact U . Ainsi, la représentation d'un système (e.g. une commande) par un ensemble de règles qualitatives est validée.

Néanmoins, il reste le problème du choix des paramètres du système flou ainsi créé. Au vu de ces propriétés d'approximation, il est concevable de développer une commande floue adaptative dont l'objectif est alors d'approximer la commande "idéale" du système considéré. Wang a développé une loi d'adaptation des paramètres de sortie (qui sont, dans la pratique, les plus délicats à régler) assurant la stabilité d'un système non-linéaire. L'inconvénient de cette méthode est qu'il devient vite difficile d'interpréter les effets de la commande floue.

Nos travaux ont consisté à étendre la démonstration des propriétés d'approximation d'un système flou en considérant une approche logique, et non pas statistique, lors de l'agrégation

des différentes règles floues. Avec cette approche, la forme de la base de règles initiale est conservée et il est par conséquent possible d'étudier de manière qualitative les réponses de la commande floue. Ceci permet à l'opérateur de garder la maîtrise du système de commande.

Approche multi-modèle

Les automaticiens ont développé de nombreux types de commandes répondant à des critères de performances particuliers. Néanmoins, la représentation par un modèle global d'un système complexe peut rendre difficile la réalisation de ces commandes. De plus, une représentation multi-modèle d'un système complexe peut nécessiter la définition de commandes hétérogènes (mise en place de la commande la plus performante dans un domaine particulier). L'objectif de la commande multi-modèle développée dans ce mémoire a été de proposer une méthodologie utilisant les propriétés d'interpolation de la logique floue afin d'unifier des commandes hétérogènes réalisées à partir de la connaissance de modèles simples du système.

Les exemples de commande multi-modèle utilisant des modèles locaux ont montré l'intérêt qu'offre ce type d'approche. Néanmoins, il est nécessaire de réaliser des identifications locales du système.

Lorsque cela s'avère impossible, l'utilisation de modèles génériques offre la possibilité de représenter dans la base des modèles choisis et sur la trajectoire de commande, le système complexe étudié.

Les perspectives de ces travaux sont de parvenir à définir le domaine de stabilité structurelle de la commande multi-modèle, et dans le cas de l'utilisation de modèles génériques d'utiliser la modélisation effectuée pour améliorer les performances de la commande (adaptation des commandes élémentaires, modification des modèles, ...).

Annexes

Annexe 1 : Démonstration du théorème de Wang

Théorème [Wan 1992a]

Soit Y , l'ensemble des fonctions caractérisant un système flou S de la forme :

$$f_s(x) = \frac{\sum_{i=1}^m y^i \prod_{j=1}^n \mu_{X_j^i}(x_j)}{\sum_{i=1}^m \prod_{j=1}^n \mu_{X_j^i}(x_j)}, \quad (\text{A1.1})$$

où $\mu_{X_j^i}$, $i = 1$ à m , $j = 1$ à n , est la fonction d'appartenance du sous-ensemble flou X_j^i de l'univers de discours U_j et les $y^i \in \mathfrak{R}$ des constantes. Alors pour toute fonction continue f de l'ensemble compact $U = U_1 \times \dots \times U_n \subset \mathfrak{R}^n$ et pour toute constante arbitraire ε positif, il existe une fonction f_s de Y telle que :

$$\sup_{x \in U} |f(x) - f_s(x)| \leq \varepsilon. \quad (\text{A1.2})$$

Démonstration

La démonstration de ce théorème est basée sur le théorème de Stone-Weierstrass [Rud 1976].

Théorème de Stone-Weierstrass :

Soit Z un ensemble de fonctions continues à valeurs réelles d'un ensemble compact U et d_∞ une distance métrique définie par :

$$d_\infty(f_1, f_2) = \sup_{x \in U} (|f_1(x) - f_2(x)|). \quad (\text{A1.3})$$

Si Z est une algèbre, c'est-à-dire si :

$$\begin{aligned} \forall f_1 \text{ et } f_2 \in Z, \\ f_1 + f_2 \in Z, \\ f_1 f_2 \in Z, \\ \forall \lambda \in \mathfrak{R}, \lambda f_1 \in Z, \end{aligned} \quad (\text{A1.4})$$

si de plus :

$$\forall (x, y) \in U^2, \text{ avec } x \neq y, \exists f \in Z / f(x) \neq f(y), \quad (\text{A1.5})$$

et si enfin :

$$\forall x \in U, \exists f \in Z / f(x) \neq 0, \quad (\text{A1.6})$$

alors, (Z, d_∞) est dense dans $(C[U], d_\infty)$.

Alors, soit deux fonctions f_1 et f_2 de Y :

$$f_1(x) = \frac{\sum_{i=1}^{m1} y1^i \prod_{j=1}^n \mu_{x1j}(x_j)}{\sum_{i=1}^{m1} \prod_{j=1}^n \mu_{x1j}(x_j)}, \quad (\text{A1.7})$$

$$f_2(x) = \frac{\sum_{i=1}^{m2} y2^i \prod_{j=1}^n \mu_{x2j}(x_j)}{\sum_{i=1}^{m2} \prod_{j=1}^n \mu_{x2j}(x_j)}. \quad (\text{A1.8})$$

L'expression de la somme de ces deux fonctions est :

$$f_1(x) + f_2(x) = \frac{\sum_{i1=1}^{m1} \sum_{i2=1}^{m2} (y1^{i1} + y2^{i2}) \prod_{j=1}^n \mu_{x1j^{i1}}(x_j) \mu_{x1j^{i2}}(x_j)}{\sum_{i1=1}^{m1} \sum_{i2=1}^{m2} \left(\prod_{j=1}^n \mu_{x1j^{i1}}(x_j) \mu_{x1j^{i2}}(x_j) \right)} \in Y. \quad (A1.9)$$

Le produit de ces deux fonctions est défini par

$$f_1(x)f_2(x) = \frac{\sum_{i1=1}^{m1} \sum_{i2=1}^{m2} (y1^{i1} y2^{i2}) \prod_{j=1}^n \mu_{x1j^{i1}}(x_j) \mu_{x1j^{i2}}(x_j)}{\sum_{i1=1}^{m1} \sum_{i2=1}^{m2} \left(\prod_{j=1}^n \mu_{x1j^{i1}}(x_j) \mu_{x1j^{i2}}(x_j) \right)} \in Y. \quad (A1.10)$$

De plus :

$$\forall \lambda \in \mathfrak{R}, \lambda f_1(x) = \frac{\sum_{i=1}^{m1} \lambda y1^i \prod_{j=1}^n \mu_{x1j^i}(x_j)}{\sum_{i=1}^{m1} \prod_{j=1}^n \mu_{x1j^i}(x_j)} \in Y. \quad (A1.11)$$

Par conséquent, (Y, d_∞) est une algèbre. Pour monter (A1.5), il suffit de trouver une fonction f satisfaisant cette condition. Wang construit cette fonction à partir de fonctions d'appartenance de forme gaussienne (A1.12) et de deux règles floues.

$$\mu_{x_i}(x) = \exp\left(-\frac{(x - x_i)^2}{2}\right) \quad (A1.12)$$

Nous proposons ici une autre méthode de construction. Soit deux vecteurs quelconques :

$$x^0 = [x^0_1, \dots, x^0_n]^T \in U^n, \quad (A1.13)$$

$$y^0 = [y^0_1, \dots, y^0_n]^T \in U^n. \quad (A1.14)$$

On considère deux règles floues :

$$R^i : \text{SI } x_1 \text{ est } X_1^i \text{ ET } \dots \text{ ET } x_n \text{ est } X_n^i \text{ ALORS } z \text{ est } Z^i. \quad (A1.15)$$

On construit les sous-ensembles flous X_j^i avec pour unique contrainte :

$$\begin{aligned}\mu_{x_i^1}(x_i) &= 1 \text{ ssi } x_i = x_i^0, \\ \mu_{x_i^2}(x_i) &= 1 \forall x_i \in U_i.\end{aligned}\tag{A1.16}$$

Alors pour une fonction f de la forme (A1.7) :

$$f(x^0) = \frac{z^1 + z^2}{2},\tag{A1.17}$$

$$f(y^0) = \frac{z^2 + z^1 \prod_{j=1}^n \mu_{x_j^1}(y_j^0)}{1 + \prod_{j=1}^n \mu_{x_j^1}(y_j^0)}.\tag{A1.18}$$

Comme $\prod_{j=1}^n \mu_{x_j^1}(y_j^0) < 1$ ($= 1$ ssi $y^0 = x^0$), pour $z^1 \neq z^2$, $f(x^0) \neq f(y^0)$.

Enfin la forme de l'équation (A1.17), avec la condition $z^1 \neq z^2$ assurée (A1.6).

Annexe 2 : Démonstration du théorème d'approximation

Théorème

Soit Y l'ensemble des fonctions caractérisant un système flou de la forme :

$$f_S(\mathbf{x}) = \frac{\sum_{i=1}^r y_i \mu_Y(y_i)}{\sum_{i=1}^r \mu_Y(y_i)} = \frac{\sum_{i=1}^r y_i \bigvee_{j=1}^{r_i} \left(\mu_{X_1^{i,j}}(x_1) \wedge_2 \dots \wedge_2 \mu_{X_n^{i,j}}(x_n) \right)}{\sum_{i=1}^r \bigvee_{j=1}^{r_i} \left(\mu_{X_1^{i,j}}(x_1) \wedge_2 \dots \wedge_2 \mu_{X_n^{i,j}}(x_n) \right)}, \quad (\text{A2.1})$$

où $\mu_{X_k^{i,j}}$, $i = 1$ à r , $j = 1$ à r_i , $k = 1$ à n , est la fonction d'appartenance du sous-ensemble flou $X_k^{i,j}$ de l'univers de discours U_j ; $y_i \in \mathfrak{R}$, $i = 1$ à r , des constantes; \wedge_2 et \bigvee sont respectivement une T-norme et une T-conorme. Alors pour toute fonction continue f de l'ensemble compact $U = U_1 \times \dots \times U_n \subset \mathfrak{R}^n$ et pour toute constante arbitraire ε positif, il existe une fonction f_S de Y telle que :

$$\sup_{\mathbf{x} \in U} |f(\mathbf{x}) - f_S(\mathbf{x})| \leq \varepsilon. \quad (\text{A2.2})$$

Démonstration

Comme U est un ensemble compact, il existe un recouvrement de U par des hypercubes P_i .

La figure ci-dessous présente ce type de recouvrement, pour un ensemble U de dimension 2.

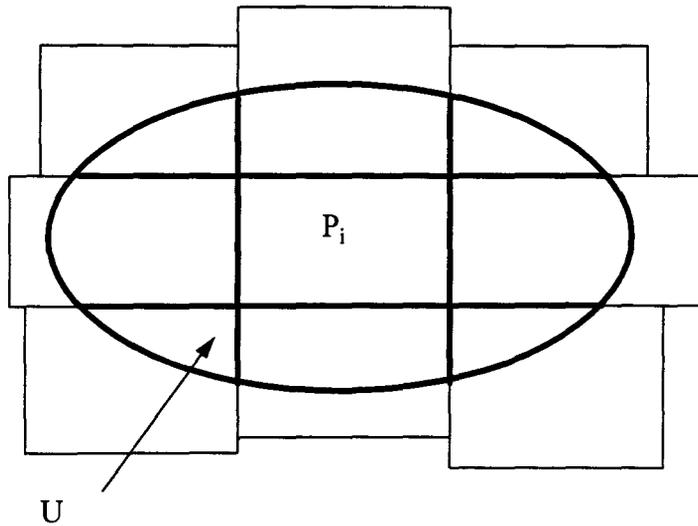


Figure A2.1 : Partition de U par les hypercubes P_i .

Remarque :

L'utilisation, lors de la partition floue de U, de fonctions d'appartenance de forme trapézoïdale telles que celles présentées sur la figure suivante assure ce type de recouvrement.

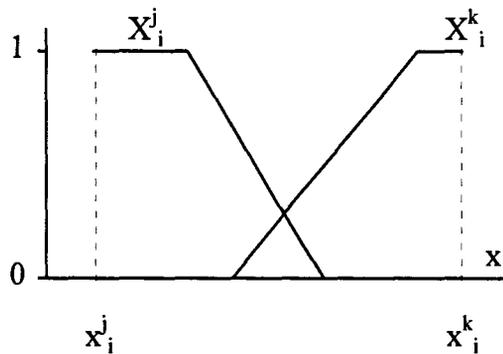


Figure A2.2 : Exemple de fonctions d'appartenance trapézoïdales.

La fonction f est uniformément continue dans l'ensemble compact U. Par conséquent, il existe une partition de l'ensemble U par les hypercubes P_i telle que :

$$\forall (x, y) \in P_i, \|f(x) - f(y)\| \leq \epsilon. \tag{A2.3}$$

Soit alors le système flou où la partition de l'ensemble U par des fonctions d'appartenance trapézoïdales est réalisée de façon à générer les hypercubes P_i . De plus à chaque sommet de l'hypercubes P_i correspond une règle R^i décrite par

$$R^i : \text{SI } x \text{ est } X^i \text{ ALORS } y \text{ est } Y^i, \tag{A2.4}$$

où Y^i est un singleton flou avec $y^i = f(x^i)$, x^i correspondant à l'un des 2^n sommets de l'hypercube P_i . Alors dans ce cas, pour tout x appartenant à un hypercube P_i , uniquement les règles relatives aux sommets de P_i sont actives. La sortie du système flou est alors :

$$f_s(x) = \frac{\sum_i y^i \mu_{Y^i}(x)}{\sum_i \mu_{Y^i}(x)}, \quad (\text{A2.5})$$

où $0 \leq \mu_{Y^i}(x) \leq 1$ correspond à l'union des règles donnant la même conclusion y^i . Par conséquent, le point $(x, f_s(x))$ est contenu dans l'hypervolume défini par les points (x^i, y^i) et donc :

$$\forall x \in P_i, |f(x) - f_s(x)| \leq \varepsilon. \quad (\text{A2.6})$$

Cette relation est vérifiée pour tous les hypercubes P_i et par conséquent :

$$\sup_{x \in U} |f(x) - f_s(x)| \leq \varepsilon. \quad (\text{A2.7})$$

BIBLIOGRAPHIE

- [Buc 1992] J.J. Buckley, "Universal fuzzy controllers", *Automatica*, Vol. 28 n°6 pp 1245-1248, 1992.
- [Bor 1990] P. Borne, G. Dauphin-Tanguy, J.P. Richard, F. Rotella, I. Zambettakis, "Commande et optimisation des processus", Editions Technip, 1990.
- [Del 1995a] F. Delmotte, L. Dubois, A.M. Desodt, P. Borne, "Using trust in uncertainty theories", soumis à *Information & Systems Engineering*.
- [Del 1995b] F. Delmotte, L. Dubois, P. Borne, "A general scheme for multi-model controller", à paraître dans *International Journal of Mathematics and Computers in Simulation*, 1995.
- [Dub 1994] L. Dubois, J.Y. Dieulot, P. Borne, "Use of fuzzy logic in multi-model control", *IEEE International Conference on System, Man and Cybernetics*, Proc. Vol 1 pp 371-374, Octobre 1994, San-Antonio (USA, Texas).
- [Dub 1995a] L. Dubois, F. Delmotte, J.Y. Dieulot, P. Borne, "Improving genetic algorithm using fuzzy logic", 5th *International IMACS-Symposium on Systems Analysis and Simulation*, publié dans *Systems Analysis Modelling and Simulations*, Vol. 18-19, pp 241-244, Juin 1995, Berlin (Allemagne).
- [Dub 1995b] L. Dubois, F. Delmotte, P. Borne, "A quasi-optimization method of control for ill-defined system using a multi-model approach", Septembre 1995, *Miedzyzdroje* (Pologne).
- [Ish 1993] A. Ishigame, T. Furukawa, S. Kawamoto, T. Taniguchi, "Sliding mode controller design based on fuzzy inference for nonlinear systems", *IEEE Transactions on Industrial Electronics*, Vol. 40 n°1 pp 64-70, 1993.
- [Joh 1994] Tor A. Johansen, "Fuzzy model based control: stability, robustness and performance issues", *IEEE Transactions on fuzzy system*, Vol. 2 n°3 pp 221-234, 1994.
- [Lee 1990] C.C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller" Part I and II, *IEEE Transactions on System, Man and Cybernetics*, Vol. 20 n°2 pp 404-435, 1990.
- [Man 1974] E.H. Mandani, "Application of fuzzy algoritms for simple dynamic plant", *Proceeding IEE*, Vol. 121 n°12 pp 1585-1588, 1974.
- [Ous 1991] A. Oustaloup, "La commande CRONE", Editions Hermes, 1991.

- [Ous 1994] A. Oustaloup, "La robustesse : Analyse et synthèse de commandes robustes", Editions Hermes, 1994.
- [Per 1994] W. Perruquetti, "Commande à modes glissants", Rapport interne, LAIL URA CNRS D1440, Ecole Centrale de Lille, Villeneuve d'Ascq, 1994.
- [Rud 1976] W. Rudin, "Principles of Mathematical Analysis", New-York : Mc Graw-Hill, 1976.
- [Slo 1983] J.J. Slotine, S.S. Satry, "Tracking control problem of non-linear systems using sliding surfaces with application to robot manipulators", International Journal of Control, n° 38 pp 465-492.
- [Tak 1985] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modelling and control", IEEE Transactions on System, Man and Cybernetics, Vol. 15 pp 110-132, 1985.
- [Utk 1965] V.I. Utkin, "On compensation of the forced term of motion in variable-structure control systems", Izv. AN SSSR, "Tekhnicheskaya Kibernetika" series, n° 4 pp 169-173, 1965.
- [Utk 1992] V.I. Utkin, "Sliding Modes in control and optimization", Springer-Verlag, 1992.
- [Wan 1992a] L.X. Wang, J. Mendel, "Fuzzy Basis Functions, Universal Approximation, and orthogonal Least-Squares Learning", IEEE Transactions on Neural Networks, Vol. 3 n°5 pp 807-814, 1992.
- [Wan 1992b] L.X. Wang, J. Mendel, "Generating fuzzy rules by learning from examples", IEEE Transactions on Systems, Man and Cybernetics, Vol. 22 pp 1414-1427, 1992.
- [Wan 1993] L.X. Wang, "Stable Adaptive Fuzzy Control of Nonlinear Systems", IEEE Transactions on Fuzzy Systems, Vol. 1 n°2 pp 146-155, 1993.
- [Web 1983] S. Weber, "A general concept of fuzzy connectives, negations and implications based on t-norms and t-comorms", Fuzzy Sets and Systems, Vol. 11, pp 115-134, 1983.
- [Zad 1965] L.A. Zadeh, "Fuzzy sets", Informatic control, Vol. 8 pp 338-353, 1965.
- [Zad 1973] L.A. Zadeh, "Outline of a new approach to the analysis complex systems and decision processes", IEEE Transactions on System, Man and Cybernetics, Vol. 3 pp 28-44, 1973.
- [Zen 1994] X.J. Zeng, M.G. Singh, "Approximation Theory of Fuzzy Systems - SISO Case", IEEE Transactions on Fuzzy Systems, Vol 2 n°2 pp 162-175, 1994.