



M0021/01B

المدرسة الوطنية المتعددة الفنون
Ecole Nationale Polytechnique

Département: Electronique

المدرسة الوطنية المتعددة الفنون
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Thèse de Magister

Option : Systèmes de Traitement de l'Information

Présentée par

Yamine SELLAMI

Ingénieur d'Etat en Electronique,

Option : Contrôle, Univ. Blida

Thème

*Contribution à la modélisation floue des systèmes
Application à la commande d'un moteur asynchrone*

Soutenue publiquement en juin 2001, devant le jury composé de :

Président :	M. HADDADI	Maitre de conférence	à l'ENP
Rapporteurs :	R. OUIGUINI	Maitre de recherche	au CDTA
	C. LARBES	PhD, Chargé de cours	à l'ENP
Examineurs :	M.S. BOUCHERIT	Maitre de conférence	à l'ENP
	M.S. AIT-CHEIKH	Chargé de cours	à l'ENP
	E.M. BERKOUK	Maitre de conférence	à l'ENP



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

Département: Electronique

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

Thèse de Magister

Option : Systèmes de Traitement de l'Information

Présentée par

Yamine SELLAMI

Ingénieur d'Etat en Electronique,

Option : Contrôle, Univ. Blida

Thème

*Contribution à la modélisation floue des systèmes
Application à la commande d'un moteur asynchrone*

Soutenue publiquement en juin 2001, devant le jury composé de :

Président :	M. HADDADI	Maître de conférence	à l'ENP
Rapporteurs :	R. OUIGUINI	Maître de recherche	au CDTA
	C. LARBES	PhD, Chargé de cours	à l'ENP
Examineurs :	M.S. BOUCHERIT	Maître de conférence	à l'ENP
	M.S. AIT-CHEIKH	Chargé de cours	à l'ENP
	E.M. BERKOUK	Maître de conférence	à l'ENP

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ
بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

المدرسة الوطنية المتعددة التقنيات
BIBLIOTHEQUE — المكتبة
Ecole Nationale Polytechnique

C'est

À mon Bon Dieu

À mes parents, frères et sœurs

*À tous mes amis et tous ceux qui m'aiment, notamment,
Djamel, Mokhtar, Linda, Noureddine, Fidy, Salvador,
Sofiane, J.Pierre, Zobeir, Alain, Mohammed...*

Aux chers lecteurs

À Nacéra et Mami Noël

Que je dédie ce modeste travail

Remerciements



Cette thèse est le fruit de beaucoup d'efforts et de passionnantes recherches. Je souhaite qu'elle présentera quelques utilités pour ceux qui veulent avancer dans le même sens de recherche.

J'ai pu m'engager dans cette voie que parce que j'ai trouvé aux laboratoires de l'École nationale polytechnique (ENP) et à ceux du Centre de développement et des technologies avancées (CDTA) un accueil de qualité ; comme j'ai bénéficié d'un soutien exceptionnel auprès du Centre d'études diocésain des Glycines (Alger).

Je tiens à exprimer toute ma gratitude à mes promoteurs Mrs R. Ouiguni et C. Larbes. J'aimerais pouvoir mentionner aussi Mlle L. Barazane qui a bien voulu superviser mon projet. J'ai apprécié l'attention et l'intérêt qu'ils ont accordé à mon travail. Leurs efforts constants pour assurer de bonnes conditions matérielles et scientifiques, leur ouverture et leur disponibilité ont été remarquables, créant un environnement propice à la recherche et à la créativité personnelle.

Nombreux sont ceux qui ont contribué à la mise en œuvre de la présente thèse. Merci à tous et particulièrement aux P. G. Nicolas, J. Fisset et J.P. Voreux, à M. Cheikh, M.N. Aouamour, A. Mezian, M. Bendara, à mes oncles Mohammed et Rachid et à mon correspondant Mounir Ben Ghalia.

J'adresse mes vifs remerciements à Messieurs M. Haddadi, M.S. Ait-Cheikh, M.S. Boucherit et E.M. Berkouk qui ont bien voulu me faire l'honneur de lire et de juger mes travaux.

Je tiens encore à remercier les personnels de la bibliothèque du CDTA, du Centre de documentation de l'ENP, de la bibliothèque de l'INELEC (Univ. Boumerdes) et de la bibliothèque de recherche du centre des Glycines (Alger) pour l'aide précieuse qu'ils m'ont offerte.

Merci enfin à mes anciens professeurs et à tous ceux qui, d'une façon ou d'une autre, m'ont accordé leur soutien, qu'il soit matériel, intellectuel ou moral.

Résumé

La commande des systèmes complexes et incertains, entre autres les moteurs asynchrones, est un problème contrariant pour les chercheurs. De ce fait, plusieurs stratégies de commande linéaire et non-linéaire ont vu le jour pour résoudre les problèmes de découplage, de robustesse, de simplicité de réglage et de mise en œuvre. La commande par logique floue, vu sa capacité, constitue une bonne solution pour ce genre de problèmes.

Dans ce contexte, nous avons étudié différents aspects de la logique floue, des contrôleurs flous et des modèles flous, ainsi que leur introduction dans la commande des systèmes. Nous avons axé notre travail sur une technique de modélisation floue, servant à imiter le système par une représentation floue tout en tenant compte de la description analytique grossière et des incertitudes du système. Deux commandes ont tiré profit de cette technique :

- ♦ Une commande par flux orienté basée sur un modèle inverse flou,
- ♦ Une commande floue robuste à structure variable basée sur un modèle flou du moteur.

Mots clés : Modélisation floue, commande basée sur un modèle flou, retour d'état non-linéaire, commande par flux orienté, commande à structure variable, commande à structure variable floue, modèle inverse flou.

Abstract

The control of complex and uncertain dynamical systems, particularly the induction motor drives, is a problem for the researchers. Thus, several linear and non-linear control strategies have appeared to solve the decoupling, robustness and practical problems. Fuzzy logic control is a good solution to such problems.

In this field, we have studied different aspects of fuzzy logic, fuzzy controllers and fuzzy models, even their introduction in control systems. We have pointed our work in a technique of fuzzy modelling, which help to imitate the system model by a linguistic fuzzy rule base representation, with taking into account the mathematical description and the boundary uncertainties of the system. Two control systems have profited from this technique:

- ♦ Control based on an inverse fuzzy model with field orientation,
- ♦ Fuzzy robust control based on a linguistic fuzzy model of the motor.

Key words: Fuzzy modelling, fuzzy model-based control, non-linear state feedback, field oriented control, variable structure control, fuzzy variable structure control, inverse fuzzy model.

ملخص

إن التحكم في الأنظمة المعقدة وغير الدقيقة، لاسيما المحركات الكهربائية اللامتزامنة الثلاثية الأطوار، يعتبر مشكلة تواجه الباحثين. وإنطلاقاً من ذلك، برزت عدة إستراتيجيات للتحكم الخطي وغير الخطي من أجل حل مشاكل الفصل بين المتغيرات الداخلية، وعدم التأثير بعوامل الاضطرابات والإرتيابات، وبساطة الضبط والتشغيل. ويأتي التحكم بواسطة المنطق الغامض كحل جيد لمثل هذه المشاكل، وذلك بفضل قدراته الخاصة.

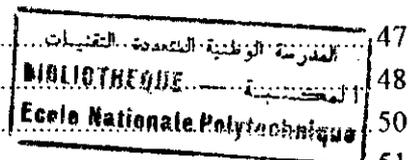
وفي هذا الإطار، قمنا بدراسة عدة جوانب للمنطق والمراقبات والنماذج الغامضة، بالإضافة إلى إدخال هذه الأخيرة على أنظمة التحكم. وقد زكنا عملنا على تقنية لتشكيل النماذج الغامضة، تسمح بتمثيل غامض للنظام أخذاً بعين الاعتبار وصفه التحليلي (الرياضي) العام وإرتيابه. حيث طبقنا هذه التقنية في إعداد شكلين من التحكم، هما:

- ♦ تحكم بتدفق موجه قائم على نموذج عكسي غامض،
- ♦ تحكم غامض غير متأثر بالاضطرابات والإرتيابات وذات هيكل متغير قائم على نموذج غامض للنظام.

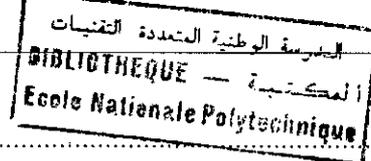
الكلمات المفتاحية: تشكيل غامض للنماذج، تحكم قائم على نموذج غامض، تغذية إسترجاعية غير خطية، تحكم بتدفق موجه، تحكم ذات هيكل متغير، تحكم غامض ذات هيكل متغير، نموذج عكسي غامض.

Table des matières

TABLE DES MATIÈRES	i
TABLE DES FIGURES.....	v
NOMENCLATURE	ix
INTRODUCTION GÉNÉRALE	1
I.1 Principales contributions de la thèse	2
I.2 Organisation de la thèse	3
I. MODÉLISATION ANALYTIQUE DE LA MACHINE ASYNCHRONE.....	5
I.1. Introduction	5
I.2. Mise en équations de la machine asynchrone triphasée	6
I.3. Transformation de Park	7
I.4. Équations de la machine biphasée équivalente	9
I.5. Définition des différents référentiels.....	10
I.6. Modélisation de l'association convertisseur - moteur asynchrone.....	11
I.7. Représentation d'état du système	13
I.8. Représentation du système sous forme canonique	16
Étape 1. Choix des grandeurs de sortie.....	16
Étape 2. Calcul du degré relatif	16
Étape 3. Choix de la transformation $\phi(x)$	17
Étape 4. Retour d'état non-linéaire.....	18
I.9. Conclusion.....	20
II. DIFFÉRENTES STRATÉGIES DE COMMANDE DU MAS.....	21
II.1. Introduction.....	21
II.2. Commande à flux orienté (CFO)	22
II.2.1. Résultats de simulation	25
Test de robustesse.....	25
II.3. Commande par placement de pôles basée sur un retour d'état linéarisant	33
II.3.1. Résultats de simulation	34
II.4. Commande à structure variable	43
II.4.1. Conception des régulateurs à structures variables.....	43
II.4.1.1. Choix de la surface de glissement	44
II.4.1.2. Choix des grandeurs de commande.....	44
II.4.1.3. Condition d'existence du mode de glissement	45
II.4.1.4. La commande équivalente	45



II.4.2. Cas des systèmes mis sous forme canonique	47
II.4.3. Inconvénients de la CSV	48
II.4.4. Réduction du phénomène de chattering	50
II.4.5. Application au moteur asynchrone	51
II.4.6. Résultats de simulations	52
II.5. Commande par logique floue	59
II.5.1. Introduction	59
II.5.2. Théorie des ensembles flous	59
II.5.2.1. Quelques caractéristiques des ensembles flous	60
II.5.2.2. Fonctions d'appartenance	61
II.5.2.3. Opérations de base sur les ensembles flous	62
II.5.2.4. Produit cartésien des ensembles flous	62
II.5.3. Variable linguistique	63
II.5.4. Principe d'extension	63
II.5.5. Les systèmes d'inférence flous	64
II.5.5.1. Configuration de base d'un système d'inférence flou	65
II.5.6. Les contrôleurs flous	69
II.5.6.1. Différents types de contrôleurs flous	69
a) Contrôleurs PI-flou et PD-flou	69
b) Contrôleurs adaptatifs flous	70
c) Commande à structure variable floue	71
II.5.7. Conception d'un contrôleur flou	72
II.5.7.1. Cas de l'absence du modèle du système	74
II.5.7.2. Cas de la présence du modèle analytique du système	74
II.5.7.3. A partir du modèle flou du système	75
II.6. Conclusion	76
III. MODELISATION FLOUE DES SYSTEMES INCERTAINS	77
III.1. Introduction	77
III.2. Évaluation des systèmes réels et flous	78
III.3. Les modèles flous à base de règles	80
III.3.1. Partitionnement flou de l'espace d'entrée	81
III.3.1.1. Partitionnement homogène	81
III.3.1.2. Partitionnement par concaténation d'ensembles flous	83
III.3.1.3. Partitionnement optimal	84
III.3.2. Différents types de modèles flous	84
III.3.2.1. Le modèle flou linguistique	85
Mécanisme d'inférence	85
Défuzzification	87
III.3.2.2. Modèle flou relationnel	89
Mécanisme d'inférence	90
III.3.2.3. Modèle flou de Takagi-Sugeno	91
Mécanisme d'inférence	93



III.3.2.4. Modèle flou de Tsukamoto.....	94
III.4. Élaboration d'un modèle flou.....	95
III.5. Modélisation floue linguistique des systèmes incertains.....	96
III.5.1. Condition d'existence de modèle flou selon la méthode proposée.....	97
III.5.2. Algorithme proposé de la modélisation floue.....	97
Étape 1. Partitionnement de l'espace d'entrée.....	98
Étape 2. Construction des intervalles des fonctions f et g	99
Étape 3. Définition des ensembles flous des sorties.....	100
Étape 4. Dédution de la base de règles.....	101
III.5.3. Évaluation de la base de règles du vecteur de commande.....	103
III.5.4. Obtention du modèle flou.....	103
III.6. Conclusion.....	104
IV. COMMANDE FLOUE BASEE SUR UN MODELE FLOU.....	105
IV.1. Introduction.....	105
IV.2. Commande par un modèle inverse flou.....	107
IV.2.1. Élaboration du modèle inverse flou par l'inversion d'une base de règles.....	108
IV.2.2. Elaboration du modèle inverse flou à partir d'une connaissance analytique du modèle inverse.....	110
IV.2.3. Application au moteur asynchrone.....	111
IV.2.4. Décomposition du modèle inverse flou.....	113
IV.2.5. Résultats de simulation.....	114
IV.3. Commande à structure variable floue.....	121
IV.3.1. Algorithme de l'élaboration de la CSV floue.....	123
Étape 1. Construction du vecteur flou d'erreur $F_e^{(k_1, \dots, k_n)}$	123
Étape 2. Construction du vecteur flou de commutation $F_s^{(k_1, \dots, k_n)}$	123
Étape 3. Choix de la dynamique floue désirée $F_s^{(k_1, \dots, k_n)}$	124
Étape 4. Calcul de la loi de commande floue $F_u^{(k_1, \dots, k_n)}$	126
IV.3.2. Cas des systèmes mis sous forme canonique.....	127
IV.3.2.1. Objectif de la commande.....	128
IV.3.2.2. Modélisation floue du sous-système.....	128
IV.3.2.3. Élaboration de la CSV floue.....	129
Étape 1. Construction du vecteur d'erreur.....	129
Étape 2. Construction de la fonction floue de commutation.....	130
Étape 3. Choix de la dynamique floue désirée $F_s^{(k_1, \dots, k_r)}$	130
Étape 4. Calcul de la loi de commande $F_u^{(k_1, \dots, k_r)}$	131
IV.3.3. Application à la commande du moteur asynchrone.....	132
IV.3.3.1. Introduction.....	132
IV.3.3.2. Conception du modèle flou du moteur asynchrone.....	132
Algorithme de la modélisation floue :.....	133
Étape 1. Partitionnement de l'espace d'entrée.....	133
Étape 2. Construction des intervalles des fonctions f_1 , g_1 et g_2	133

Étape 3. Définition des ensembles flous des sorties	134
Étape 4. Déduire la base de règles	134
IV.3.3.3. Elaboration de la commande floue robuste	135
Étape 1. Construction des variables d'erreur floues	135
Étape 2. Construction des fonctions floues de commutation	136
Étape 3. Choix de la dynamique floue désirée	136
Étape 4. Calcul de la loi de commande floue	137
IV.3.4. Résultats de simulation	138
IV.4. Conclusion	138
CONCLUSION GENERALE	151
Suggestions et perspectives	153
BIBLIOGRAPHIE	155
ANNEXE A. PARAMETRES DU MOTEUR ASYNCHRONE	A.1
ANNEXE B. CARACTERISTIQUES DES ENSEMBLES FLOUS	B.3
B.1. Fonctions d'appartenance usuelles	B.3
B.2. Opérateurs usuels de conjonction, disjonction et de complément	B.6
B.3. Quelques exemples des opérateurs T-norme, T-conorme et de complément	B.7
ANNEXE C. DEMONSTRATION DU LEMME (4.1)	C.9
ANNEXE D. FONCTION DE CARATHEODORY	D.11
D.1. Espace topologique	D.11
D.2. Espace métrique	D.11
D.3. Espace topologique localement compact	D.12
D.4. Fonction mesurable	D.13
D.5. Mesure au sens de Lebesgue	D.13
D.6. Mesure au sens de Radon	D.13
D.7. Fonction fortement Carathéodory	D.13

Table des figures

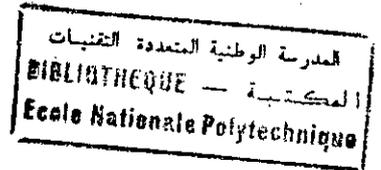


Figure 1.1 : Représentation spatiale de la transformation triphasée / biphasée	8
Figure 1.2 : Association convertisseur - moteur asynchrone	11
Figure 1.3 : Onduleur de tension	12
Figure 1.4 : Forme d'un courant de phase statorique et de la fonction logique F_r	13
Figure 1.5 : Représentation du modèle du moteur asynchrone basé sur le retour d'état non-linéaire adopté	19
Figure 2.1 : Caractéristiques mécaniques pour différentes valeurs de la fréquence d'alimentation. a) avec V_s/f_s constant, b) avec flux constant	22
Figure 2.2 : Schéma-bloc de la commande indirecte par orientation du flux rotorique.	23
Figure 2.3 : Schéma de commande indirecte par flux orienté du moteur asynchrone commandé en courant	24
Figure 2.4 : Commande par flux orienté. Comportement dynamique de la machine dans le cas idéal	27
Figure 2.5 : Commande par flux orienté. Comportement de la machine lors d'une variation de R_r de 90% à 150% (sans variation des inductances)	29
Figure 2.6 : Commande par flux orienté. Comportement de la machine lors d'une variation de R_r de 90% à 150% (avec une variation de 5% sur les inductances)	31
Figure 2.7 : Schéma de commande du moteur asynchrone basée un retour d'état linérisant	33
Figure 2.8 : Commande par placement de pôles. Comportement dynamique du moteur dans le cas idéal	35
Figure 2.9 : Commande par placement de pôles. Résultats de simulation pour différentes variations de R_r , avec estimation de la charge	37
Figure 2.10 : Commande par placement de pôles. Résultats de simulation pour différentes variations de R_r (sans estimation de la charge et avec une variation de 5% sur les inductances)	39
Figure 2.11 : Commande par placement de pôles. Résultats de simulation pour différentes erreurs sur l'estimation de C_r (avec $R_r = 150\% R_{r0}$ et une variation de 5% sur les inductances)	41
Figure 2.12 : Aspect général d'une commande à structure variable.	43
Figure 2.13 : Mise en évidence de l'écart de poursuite lors de l'implémentation pratique	49
Figure 2.14 : Réduction du phénomène du chattering par une forme linéaire de la fonction discontinue	50
Figure 2.15 : Forme générale de la commande discontinue réduisant le chattering	51

Figure 2.16 : Commande à structure variable à relais ($k=K$). Résultats de simulation pour différentes variations de R_r	53
Figure 2.17 : Commande à structure variable adoucie ($k=0$). Résultats de simulation pour différentes variations de R_r	55
Figure 2.18 : Commande à structure variable semi-adoucie ($k \neq 0$). Résultats de simulation pour différentes variations de R_r	57
Figure 2.19 : Fonction d'appartenance. a) logique classique, b) logique floue	60
Figure 2.20 : Formes usuelles des fonctions d'appartenance	61
Figure 2.21 : Exemple d'addition étendue de deux ensembles flous	64
Figure 2.22 : Structure d'un système d'inférence flou	65
Figure 2.23 : Fuzzification, a) entrée réelle, b) entrée floue	65
Figure 2.24 : Implication, a) par saturation, b) par produit	66
Figure 2.25 : Différentes méthodes d'agrégation	67
Figure 2.26 : Différentes méthodes de défuzzification	68
Figure 2.27 : Structure des régulateurs PI-flous et PD-flous	70
Figure 2.28 : Commande supervisée par logique floue	70
Figure 2.29 : Adaptation d'un contrôleur flou	71
Figure 2.30 : Génération de la commande en tenant compte des erreurs successives des variables de sortie	71
Figure 2.31 : Génération de la commande en tenant compte du calcul de la fonction de commutation	72
Figure 2.32 : Schéma de commande à structure variable floue en tenant compte du calcul de la fonction de commutation et de sa dérivée	72
Figure 2.33 : Conception d'un contrôleur flou	73
Figure 2.34 : Structure de base d'un système de commande basé sur un modèle flou	75
Figure 2.35 : Introduction de logique floue dans la commande adaptative	78
Figure 3.1 : Évaluation d'une fonction réelle, intervalle ou floue pour des arguments réels, intervalles ou flous	79
Figure 3.2 : Exemple de fonction d'appartenance gaussienne à deux dimensions	81
Figure 3.3 : Partition floue homogène de l'espace d'entrée schématisée par les noyaux des ensembles flous	82
Figure 3.4 : Partition floue obtenue par concaténation d'ensembles flous d'une partition homogène	83
Figure 3.5 : Partition floue optimale	84
Figure 3.6 : Fuzzification ou obtention du degré de validité de l'antécédent d'une règle	86

Figure 3.7 : Système d'inférence flou de Maimdani utilisant min et max pour les opérations ET et OU respectivement	88
Figure 3.8 : Modèle flou linguistique (à deux entrées et deux sorties)	90
Figure 3.9 : Modèle flou relationnel (à deux entrées et deux sorties)	90
Figure 3.10 : Système d'inférence d'un modèle linéaire flou de Takagi-Sugeno	92
Figure 3.11 : Système d'inférence d'un modèle flou de Tsukamoto	94
Figure 3.12 : Effet des incertitudes sur un système	96
Figure 3.13 : Répartition des ensembles flous pour une variable d'entrée	99
Figure 3.14 : Répartition des ensembles flous pour une fonction de sortie	100
Figure 4.1 : Structure de base d'un système de commande basé sur un modèle flou	106
Figure 4.2 : Le modèle inverse d'un système MIMO	107
Figure 4.3 : Exemple d'inversion d'une base de règles a) modèle flou original ; b) modèle inverse flou	108
Figure 4.4 : Inversion d'une base de règle de deux entrées et une sortie	109
Figure 4.5 : Schéma de commande par un modèle inverse flou remplaçant la CFO	112
Figure 4.6 : Décomposition du modèle inverse flou en des sous-systèmes flous	113
Figure 4.7 : Commande par un modèle inverse flou. Comparaison entre les deux commandes pour différentes valeurs de M_i	115
Figure 4.8 : Commande par un modèle inverse flou. Résultats de simulation pour différentes valeurs de M_i (sans variation des paramètres)	117
Figure 4.9 : Commande par un modèle inverse flou. Résultats de simulation pour différentes variations de R_r (avec un partitionnement de l'espace d'entrée de 7 ensembles flous et une variation de 5% sur les inductances)	119
Figure 4.10 : Décomposition des ensembles flous en des ensembles flous positifs et négatifs	125
Figure 4.11 : Schéma de Commande à SV floue basée sur le modèle flou du moteur asynchrone	132
Figure 4.12 : Répartition des ensembles flous pour une fonction de sortie	134
Figure 4.13 : Commande à structure variable floue. Comparaison entre la commande réelle et floue (les paramètres et le couple résistant sont connus avec certitude)	141
Figure 4.14 : Commande à structure variable floue. Comparaison entre la commande réelle et floue (les paramètres et le couple résistant sont connus avec certitude)	143
Figure 4.15 : Commande à structure variable floue. Résultats de simulation pour différentes valeurs de M_i (avec estimation de la charge et sans variation des paramètres)	145
Figure 4.16 : Commande à structure variable floue. Résultats de simulation pour différentes variations de R_r (sans estimation de charge et avec une variation de 5% sur les inductances)	147

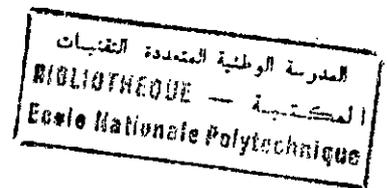
Figure 4.17 : Commande à structure variable floue. Résultats de simulation pour différentes variations de R_r (avec un partitionnement de l'espace d'entrée de 7 ensembles flous, sans estimation de charge et avec une variation de 5% sur les inductances)

149

Figure B.1 : Formes usuelles des fonctions d'appartenance



Nomenclature



c, b, c : indices correspondant aux trois phases de la machine

d, q : indices correspondant au référentiel lié au champ tournant

α, β : indices correspondant au référentiel fixe (lié au stator)

i : courant instantané

v : tension instantané

Φ : flux instantané

R_s, R_r : résistances du stator et du rotor

L_s, L_r : inductances (cycliques) du stator et du rotor

L_m : inductance mutuelle cyclique entre le stator et le rotor

L_{as}, L_{ar} : inductances d'une phase statorique et d'une phase rotorique

M_{as}, M_{ar} : inductances mutuelles entre deux phases statoriques et deux phases rotoriques

T_s : constante de temps statorique (L_s / R_s)

T_r : constante de temps rotorique (L_r / R_r)

σ : coefficient de fuite ($\sigma = 1 - L_m^2 / (L_r L_s)$)

p : nombre de paires de pôles

Ω : vitesse mécanique

ω_{sl} : vitesse de glissement

θ_r : position du rotor ($\Omega = d\theta_r / dt$)

θ : position du référentiel par rapport au stator

ω_a : vitesse de rotation du référentiel

ω_r : vitesse électrique de rotation du rotor

J : moment d'inertie

f : coefficient de frottements visqueux

C_{em} : couple électromagnétique

C_r : couple résistant

Introduction générale

Les progrès apparus dans les domaines de l'électronique, de la micro-électronique et de la micro-informatique ont permis aux machines à courant alternatifs, en particulier le moteur asynchrone à cage, de concurrencer sérieusement le moteur à courant continu à excitation séparée. Le moteur asynchrone présente divers avantages, entre autres, un faible coût, faible poids, taille réduite, construction simple, maintenance légère et bonne fiabilité de fonctionnement. Toutefois, sa commande reste complexe et difficile à mettre en œuvre. Cela est dû, en particulier, au problème de couplage étroit entre les diverses variables, et aux fortes non-linéarités que présente le modèle de la machine, au risque d'aboutir à de mauvaises performances dynamiques, comparées à celles obtenues avec le moteur à courant continu.

L'une des premières commandes apportant une solution à ce problème, est la commande vectorielle, appelée également commande par flux orienté (CFO) [Fu91][Bar93][Trz94][Igo95]. Elle a été introduite pour la première fois, en Allemagne, par Blaschke [Bla71][Bla72]. Celui-ci a présenté la commande directe, Hasse a introduit ensuite la commande indirecte. Le principe de cette méthode de commande consiste à ramener les performances du moteur asynchrone vers celles de la machine à courant continu, en orientant le vecteur flux sur un axe du référentiel lié au champ tournant. Cela a permis d'éliminer le problème de couplage entre les deux variables "couple et flux".

Néanmoins, ce type de contrôle présente un inconvénient majeur qui réside dans sa sensibilité vis-à-vis des paramètres de la machine, en particulier, de la constante de temps rotorique. En plus, les algorithmes de commande utilisés dans la commande vectorielle étaient, au début, basés sur des régulateurs classiques comme le PI conventionnel, en raison de sa simplicité et de sa stabilité [Mad96][Trz94]. Ce dernier est linéaire et dépend du modèle mathématique simplifié de la machine, lequel est déterminé pour ses paramètres physiques nominaux. Malheureusement, une variation des paramètres de la machine entraîne des incertitudes sur les coefficients du régulateur qui lui font perdre les qualités de réglage exigées. Cela est dû au fait que la robustesse des régulateurs linéaires n'est valide qu'au voisinage du point de fonctionnement, autour duquel le système est linéarisé.

Ces inconvénients sont aujourd'hui réduits par le développement de calculateurs numériques (à base de microprocesseurs, DSP, ...), par l'amélioration de la technologie des variateurs de fréquence, et par l'introduction de nouvelles techniques de contrôle telles que la commande linéarisante qui permet d'avoir un découplage entre les variables internes (couple et flux) quel que soit la vitesse de fonctionnement [Mar91][Bod94][Tut95]. Cette dernière permet l'obtention de performances comparables à celles des machines à courant continu. Malheureusement en pratique, ceci n'est pas vérifié, à cause des incertitudes inhérentes au modèle et qui dépendent de la qualité d'observation des variables d'état, ce qui a ouvert la voie au développement des observateurs linéaires et non-linéaires du flux rotorique [Mar97][Lan95][Igo95].

Dans ce contexte, plusieurs approches de commande non-linéaires robustes intègrent, lors de leur élaboration, les fluctuations paramétriques et les perturbations de charge introduites dans le moteur. Elles ont été le fruit de nombreuses recherches, parmi lesquelles nous retrouvons la commande à structure variable (CSV) [Sab89][Cha93][Sob96a][Bar98], la commande de Lyapunov [Che00], les commandes robustes H_∞ , H_2 [Bou99a]... Ces commandes présentent néanmoins une complexité dans leur mise en œuvre.

Des efforts de recherche consacrés au contrôle robuste des systèmes non linéaires incertains ont permis d'envisager une autre approche de contrôle, dite "Commande par Logique Floue" (CLF) [Zad75][Tan94][Bag96][Sin99]. Cette dernière a attiré l'attention des chercheurs durant les deux dernières décennies. Une des caractéristiques de la CLF suppose qu'on n'ait pas besoin de trouver des solutions exactes pour des systèmes complexes ; une solution approximative de la commande est suffisante. Une caractéristique importante de la théorie des ensembles flous consiste à fournir une représentation convenable des incertitudes du système [Zad65].

Traditionnellement, le contrôleur flou est conçu sans la connaissance explicite du modèle du système. Cela suit l'idée d'origine de Mamdani où les règles linguistiques sont conçues à partir des connaissances d'un expert [Mam75]. Cette approche souffre de plusieurs inconvénients :

- ☛ on ne prend pas en considération l'analyse du modèle du système,
- ☛ difficultés dans l'analyse de la stabilité globale du contrôleur flou,
- ☛ manque de prédiction sur la performance du contrôleur.

Face à ces problèmes, des algorithmes de commande basés sur des modèles flous sont apparus dans la littérature [Gha97][Sug93][Tle99]. Ces algorithmes tirent profit des résultats des travaux sur la modélisation floue des systèmes physiques [Tak85][Gha96a].

1.1 Principales contributions de la thèse

Pour bénéficier des avantages que procure la logique floue, et vu la complexité, la non-linéarité et les incertitudes du modèle du moteur asynchrone, nous avons contribué à la

modélisation floue du moteur, afin de concevoir un modèle flou susceptible d'approximer le modèle réel et de fournir une commande robuste floue.

Pour ce faire, nous proposons une reformulation du modèle du moteur. Celle-ci permet d'une part, une commande décentralisée, d'autre part, une meilleure introduction des systèmes approximatifs, en particulier les modèles flous.

Par la suite, nous effectuons une synthèse sur les différentes directions possibles de conception des régulateurs par logique floue et des modèles flous. Nous avons dirigé notre travail particulièrement sur la modélisation floue linguistique, qui est une technique basée sur une connaissance préalable de l'expression analytique du modèle, et une conversion de ce dernier en une équation différentielle floue, tout en tenant compte des incertitudes inhérentes au système.

Nous abordons ensuite deux applications de la modélisation floue : une commande par un modèle inverse flou avec orientation du flux rotorique, et une commande à structure variable floue basée sur le modèle flou linguistique du moteur.

1.2 Organisation de la thèse

Le présent travail est divisé en quatre chapitres :

- Le premier chapitre est consacré à la modélisation analytique de la machine asynchrone sous différents angles.
- Le deuxième chapitre présente l'état de l'art sur la commande d'un moteur asynchrone, en spécifiant quelques stratégies usuellement employées. Les techniques développées dans ce chapitre sont la commande par flux orienté, la commande basée sur un retour d'état linéarisant, la commande à structure variable et la commande par logique floue.
- Le troisième chapitre introduit les aspects de la modélisation floue d'un système incertain pour pallier aux imperfections de la modélisation classique.
- Dans le quatrième chapitre, nous verrons les applications de la modélisation floue en vue de la commande d'un moteur asynchrone :
 - commande à flux orienté basé sur un modèle inverse flou,
 - commande à structure variable floue basée sur le modèle flou du système.

Une conclusion générale donnera une synthèse du travail effectué, et résumera les principaux résultats obtenus, ainsi que les perspectives envisagées pour d'éventuelles améliorations.

Chapitre 1

Modélisation Analytique de la Machine Asynchrone

1.1. Introduction

La machine asynchrone a fait l'objet de nombreuses études les trois dernières décennies. Elle présente l'avantage d'être robuste, peu coûteuse, de construction simple et de maintenance réduite, en particulier lorsqu'il s'agit de la machine asynchrone à cage d'écurie. Cette dernière est la machine la plus utilisée pour obtenir de la puissance mécanique à partir du réseau alternatif; mais elle présente un système d'équations très complexe à étudier qui exige un recours aux calculs matriciels. Par suite de cette complexité, on doit développer un modèle dont le comportement dynamique soit aussi proche que possible de celui de la réalité.

Par conséquent, la théorie générale a pour but de traiter une large gamme de machines de façon unifiée, en les ramenant à un modèle unique dit "machine primitive". Ce modèle est caractérisé par un système d'axes en quadratures indicé (d,q) [Cha90], dans la mesure où l'on admet comme première approximation les hypothèses simplificatrices suivantes :

- ▣ La saturation dans le circuit magnétique est négligée, cela permet d'exprimer les flux comme fonctions linéaires des courants.
- ▣ Le circuit magnétique est parfaitement feuilleté, afin de négliger les courants de Foucault.
- ▣ Les pertes par hystérésis et effet de peau sont négligées.
- ▣ L'épaisseur de l'entrefer est considérée constante sur toute la périphérie de la machine, en négligeant l'effet des encoches.
- ▣ La force magnétomotrice créée par chacune des phases est à répartition sinusoïdale, ce qui revient à ne considérer que la fondamentale. Ce qui signifie que le flux d'enroulement à travers chaque phase et l'inductance mutuelle entre un enroulement rotorique et statorique suivent une loi sinusoïdale en fonction de l'angle rotorique.
- ▣ De même, la machine est considérée comme symétrique et équilibrée.

Dans ce chapitre, nous décrirons le modèle triphasé de la machine utilisant les hypothèses simplificatrices mentionnées ci-dessus, et le modèle biphasé équivalent.

Nous présenterons ensuite le modèle complet et réduit, dans le référentiel fixe et tournant, sous forme de représentation d'état.

Puis, nous présenterons un retour d'état linéarisant de la machine asynchrone qui consiste à appliquer une boucle interne permettant une linéarisation exacte ou partielle du système pour un cas idéal (connaissance parfaite du système) et après un changement convenable de coordonnées de l'espace d'état.

Ainsi nous mettrons l'accent sur un retour d'état non-linéaire permettant l'obtention d'un modèle mieux adapté aux systèmes approximatifs, tels que les systèmes d'inférence floue, tout en gardant décentralisée la commande du moteur.

1.2. Mise en équations de la machine asynchrone triphasée

En tenant compte des hypothèses simplificatrices et en adoptant la convention de signe moteur, les expressions générales de la machine exprimées en fonction des flux et des courants sont définies comme suit [Bar82][Cha90] :

* Équations électriques

$$\begin{aligned} [v_s] &= \frac{d}{dt} [\psi_s] + R_s [i_s] \\ [v_r] &= \frac{d}{dt} [\psi_r] + R_r [i_r] \end{aligned} \quad (1.1)$$

où $[v_s] = (v_{sa}, v_{sb}, v_{sc})^t$ et $[v_r] = (v_{ra}, v_{rb}, v_{rc})^t$ représentent les tensions des trois phases statoriques et rotoriques, respectivement.

$[i_s] = (i_{sa}, i_{sb}, i_{sc})^t$ et $[i_r] = (i_{ra}, i_{rb}, i_{rc})^t$ sont les vecteurs des courants traversant ces phases.

$[\psi_s] = (\Phi_{sa}, \Phi_{sb}, \Phi_{sc})^t$, $[\psi_r] = (\Phi_{ra}, \Phi_{rb}, \Phi_{rc})^t$ correspondent aux vecteurs des flux totalisés traversant les enroulements statoriques et rotoriques.

* Équations magnétiques :

Les expressions des flux statoriques et rotoriques sous la forme matricielle condensée s'écrivent :

$$\begin{aligned} [\psi_s] &= [L_{ss}] \cdot [i_s] + [M_{sr}] \cdot [i_r] \\ [\psi_r] &= [L_{rr}] \cdot [i_r] + [M_{rs}] \cdot [i_s] \end{aligned} \quad (1.2)$$

avec :

$$[L_{ss}] = \begin{bmatrix} L_s & M_s & M_s \\ M_s & L_s & M_s \\ M_s & M_s & L_s \end{bmatrix} ; [L_{rr}] = \begin{bmatrix} L_r & M_r & M_r \\ M_r & L_r & M_r \\ M_r & M_r & L_r \end{bmatrix}$$

$$[M_{sr}] = M_{sr} \begin{bmatrix} \cos(\theta_r) & \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) \\ \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos(\theta_r) & \cos\left(\theta_r + \frac{2\pi}{3}\right) \\ \cos\left(\theta_r + \frac{2\pi}{3}\right) & \cos\left(\theta_r - \frac{2\pi}{3}\right) & \cos(\theta_r) \end{bmatrix} = [M_{rs}]^t$$

θ_r : angle entre la phase a du stator et celle du rotor. ($\Omega = d\theta_r/dt$)

$L_s, (L_r)$: inductance propre d'une phase statorique (rotorique),

$M_s (M_r)$: inductance mutuelle entre deux phases statoriques (rotoriques).

M_{sr} : inductance mutuelle maximale entre une phase du stator et une phase du rotor.

* *Équations mécaniques :*

$$C_{em} - C_r - f_r \Omega = J \frac{d}{dt} \Omega \quad (1.3)$$

avec C_{em}, C_r, f_r, J : le couple électromagnétique, le couple résistant, le coefficient de frottement et le moment d'inertie, respectivement.

Nous aboutirons ainsi à un système de six équations différentielles et une expression du couple dont certains coefficients font intervenir des fonctions sinusoïdales dues au mouvement de rotation du rotor, d'où la complexité de la résolution analytique.

Afin de surmonter cette difficulté, on considère les enroulements biphasés équivalents aux enroulements statorique et rotorique.

1.3. Transformation de PARK

Grâce à la structure symétrique et équilibrée de la machine, la transformation de Park permet le passage du système triphasé (a, b, c) au système biphasé à deux axes fictifs (d, q) en quadrature équivalents, comme illustré à la figure 1.1.

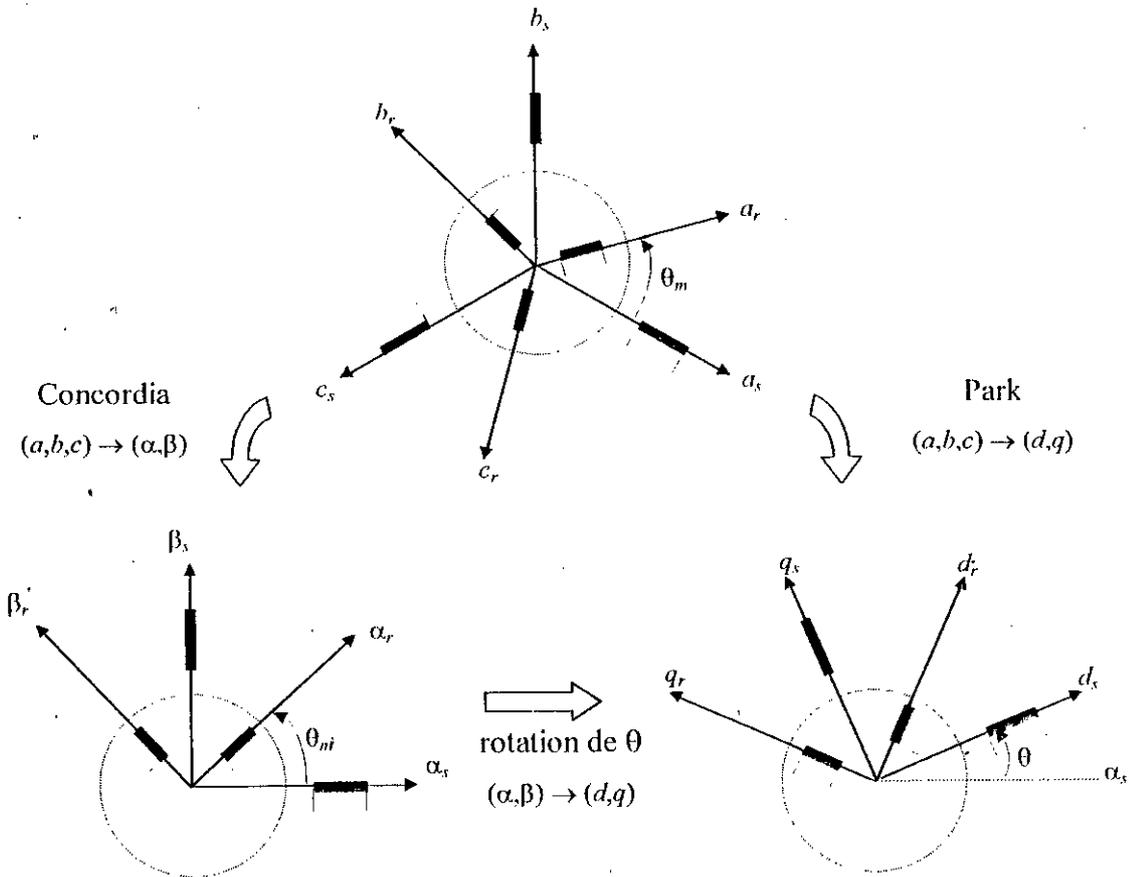


Figure 1.1 : Représentation spatiale de la transformation triphasée / biphasée

De ce fait, il est donc possible de définir une matrice $[A]$, permettant le passage des composantes X_{abc} du système triphasé aux composantes X_{dq0} du système biphasé tournant à la même vitesse, telle que :

$$[A] = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ -\sin(\theta) & -\sin\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \quad (1.4)$$

θ : étant l'angle entre la phase a du stator et l'axe d du référentiel.

Les courants, tensions et flux dans le nouveau repère sont définis comme suit :

$$\begin{bmatrix} i_d \\ i_q \\ i_o \end{bmatrix} = [A] \cdot \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} ; \quad \begin{bmatrix} v_d \\ v_q \\ v_o \end{bmatrix} = [A] \cdot \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} ; \quad \begin{bmatrix} \Phi_d \\ \Phi_q \\ \Phi_o \end{bmatrix} = [A] \cdot \begin{bmatrix} \Phi_a \\ \Phi_b \\ \Phi_c \end{bmatrix} \quad (1.5)$$

La transformation $[A]$ peut s'effectuer à partir de deux transformations successives :

– la première transforme l'enroulement triphasé en un enroulement équivalent bipolaire dans un repère fixe (α, β, o) (Figure 1.1). Elle est donnée par :

$$[A_1] = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -0.5 & -0.5 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \quad (1.6)$$

– la deuxième consiste en une rotation des axes du repère (α, β) d'un angle quelconque θ pour donner le référentiel tournant. Elle est donnée par :

$$[\Lambda_2] = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \quad (1.7)$$

1.4. Équations de la machine biphasée équivalente

Pour simplifier la représentation des équations électriques de la machine asynchrone, on a utilisé la transformation de Park, dont le but est d'arriver à rendre la matrice impédance indépendante de la variable θ_r .

Les équations électriques et magnétiques donnent alors le système suivant :

$$\begin{aligned} v_{ds} &= R_s i_{ds} + \frac{d}{dt} \Phi_{ds} - \omega_a \Phi_{qs} & v_{dr} &= R_r i_{dr} + \frac{d}{dt} \Phi_{dr} - (\omega_a - \omega_r) \Phi_{qr} \\ v_{qs} &= R_s i_{qs} + \frac{d}{dt} \Phi_{qs} + \omega_a \Phi_{ds} & v_{qr} &= R_r i_{qr} + \frac{d}{dt} \Phi_{qr} + (\omega_a - \omega_r) \Phi_{dr} \end{aligned} \quad (1.8)$$

et :

$$\begin{aligned} \Phi_{ds} &= L_s i_{ds} + L_m i_{dr} & \Phi_{dr} &= L_r i_{dr} + L_m i_{ds} \\ \Phi_{qs} &= L_s i_{qs} + L_m i_{qr} & \Phi_{qr} &= L_r i_{qr} + L_m i_{qs} \end{aligned} \quad (1.9)$$

en posant:

$\omega_a = d\theta / dt$ la vitesse de rotation du référentiel.

$\omega_r = p \cdot \Omega = p \cdot d\theta_r / dt$ la vitesse électrique de rotation du rotor.

L_s, L_r inductances propres cycliques du stator et du rotor respectivement ;

$L_s = L_{as} - M_{as}$ et $L_r = L_{ar} - M_{ar}$

L_m : inductance mutuelle cyclique entre stator et rotor ; $L_m = 3/2 M_{sr}$

Équation du couple :

$$J \frac{d}{dt} \Omega = C_{em} - C_r - f_r \Omega \quad (1.10)$$

où :

$$C_{em} = \frac{p L_m}{L_r} (\Phi_{dr} \cdot i_{qs} - \Phi_{qr} \cdot i_{ds}) \quad (1.11)$$

1.5. Définition des différents référentiels

Le référentiel est le système (d, q) associé de rotation. Dans notre cas, nous adoptons un seul référentiel pour le rotor et le stator. Il existe trois possibilités de référentiels dans la pratique. Le choix se fait en fonction du problème étudié.

Parmi les relations que nous venons de présenter, les seules qui soient affectées par le choix du référentiel sont les équations (1.8).

* Référentiel fixe par rapport au stator

Il se traduit par la condition : $\omega_d = 0$.

Les équations électriques prennent ainsi la forme suivante :

$$\begin{aligned} v_{ds} &= R_s i_{ds} + \frac{d}{dt} \Phi_{ds} & v_{dr} &= R_r i_{dr} + \frac{d}{dt} \Phi_{dr} + \omega_r \Phi_{qr} \\ v_{qs} &= R_s i_{qs} + \frac{d}{dt} \Phi_{qs} & v_{qr} &= R_r i_{qr} + \frac{d}{dt} \Phi_{qr} - \omega_r \Phi_{dr} \end{aligned} \quad (1.12)$$

Le référentiel fixe est intéressant lorsqu'on veut étudier la variation de la fréquence d'alimentation, associée ou non à la variation de la vitesse de rotation.

* Référentiel fixe par rapport au rotor

Il correspond aux transformations des grandeurs de la machine dans un référentiel tournant à la vitesse synchrone, c.-à-d. : $\omega_d = \omega_r$.

Les équations électriques sont données par :

$$\begin{aligned} v_{ds} &= R_s i_{ds} + \frac{d}{dt} \Phi_{ds} - \omega_r \Phi_{qs} & v_{dr} &= R_r i_{dr} + \frac{d}{dt} \Phi_{dr} \\ v_{qs} &= R_s i_{qs} + \frac{d}{dt} \Phi_{qs} + \omega_r \Phi_{ds} & v_{qr} &= R_r i_{qr} + \frac{d}{dt} \Phi_{qr} \end{aligned} \quad (1.13)$$

Ce référentiel est particulièrement avantageux dans l'étude des régimes transitoires où la vitesse de rotation du rotor est considérée comme constante, par exemple pour l'étude des contraintes résultant d'un court-circuit.

*** Référentiel fixe par rapport au champ tournant**

La condition qui régit ce cas est: $\omega_a = \omega_s$.

Les équations électriques du moteur sont données par :

$$\begin{aligned} v_{ds} &= R_s i_{ds} + \frac{d}{dt} \Phi_{ds} - \omega_s \Phi_{qs} & v_{dr} &= R_r i_{dr} + \frac{d}{dt} \Phi_{dr} - (\omega_s - \omega_r) \Phi_{qr} \\ v_{qs} &= R_s i_{qs} + \frac{d}{dt} \Phi_{qs} + \omega_s \Phi_{ds} & v_{qr} &= R_r i_{qr} + \frac{d}{dt} \Phi_{qr} + (\omega_s - \omega_r) \Phi_{dr} \end{aligned} \quad (1.14)$$

C'est le seul référentiel qui n'introduit pas de simplification dans les équations de la machine. Il est utilisé dans les problèmes d'alimentation des machines asynchrones par convertisseur statique de fréquence, et lorsqu'on veut étudier la fonction de transfert du moteur par rapport à de petites variations de la vitesse autour d'un régime donné [Bar82].

1.6. Modélisation de l'association convertisseur - moteur asynchrone

Le moteur asynchrone est alimenté par un onduleur de tension commandé par la stratégie "delta" qui permet une commande en courant et l'utilisation du modèle réduit du moteur (Figure 1.2).

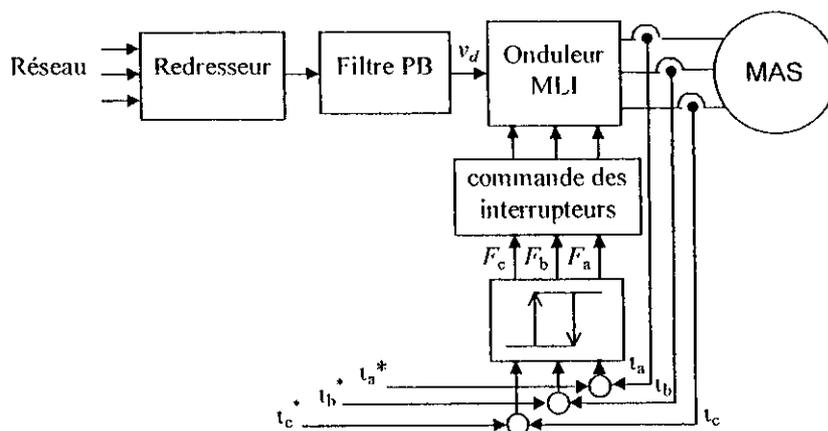


Figure 1.2 : Association convertisseur - moteur asynchrone

Les interrupteurs de l'onduleur à deux niveaux sont commandés de telle sorte que le courant de chaque phase évolue dans une bande d'hystérésis encadrant le courant de

référence correspondant. Le contrôle des courants se fait par une comparaison à hystérésis entre les courants réels et ceux de référence.

Les tensions composées aux bornes de l'onduleur sont exprimées en fonction des variables logiques F_a, F_b, F_c telle que :

$$\begin{pmatrix} v_a \\ v_b \\ v_c \end{pmatrix} = \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix} \cdot \begin{pmatrix} F_a \\ F_b \\ F_c \end{pmatrix} \cdot \frac{v_d}{3} \quad (1.15)$$

avec : v_d la tension continue fournie par redressement et filtrage de la tension triphasée du secteur.

F_i ($i = a, b, c$) représentent l'état logique des interrupteurs dont la commutation est supposée instantanée.

$$F_i = \begin{cases} 0 & \text{si } T_i \text{ conduit et } T_i' \text{ bloqué} \\ 1 & \text{si } T_i' \text{ conduit et } T_i \text{ bloqué} \end{cases}$$

T_i, T_i' ($i = a, b, c$) représentent les états des transistors de l'onduleur (Figure 1.3)

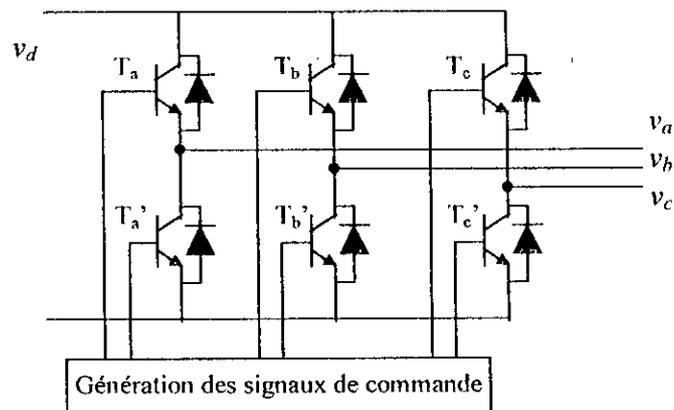


Figure 1.3 : Onduleur de tension

Le modèle du comparateur à hystérésis pour une phase est donné par :

$$F_i(k+1) = \begin{cases} 1 & \text{si } \Delta i_i > h \\ 0 & \text{si } \Delta i_i < -h \\ F_i(k) & \text{si } |\Delta i_i| \leq h \end{cases} \quad (1.16)$$

avec : $\Delta i_i = i_i - i_i^*$ ($i = a, b, c$)

h : la bande d'hystérésis

La figure 1.4 donne l'allure du courant statorique d'une phase et de la fonction logique F_i .

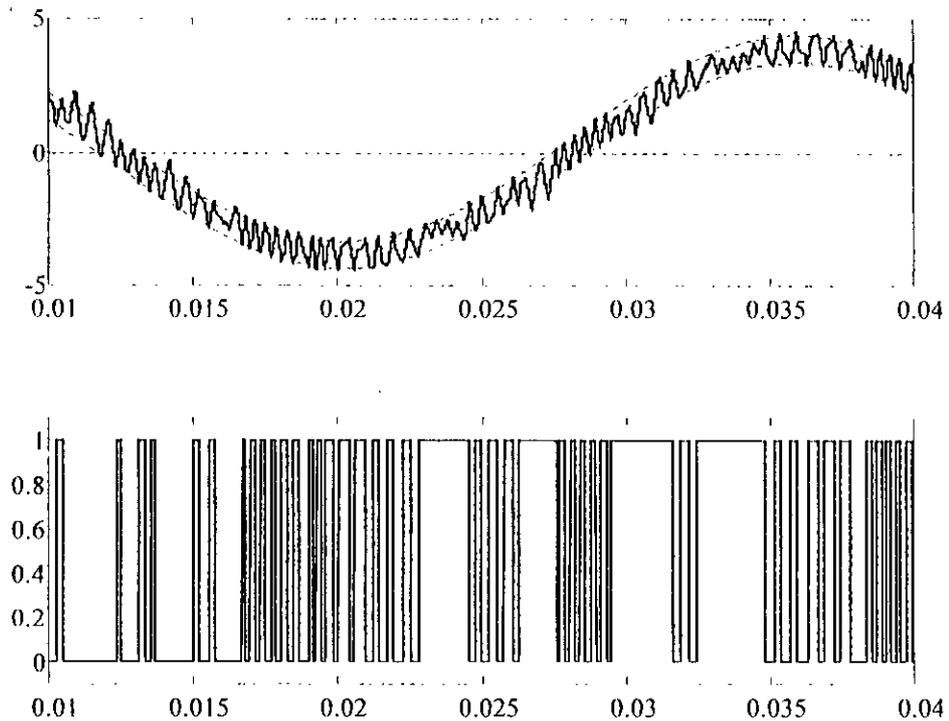


Figure 1.4 : Forme d'un courant de phase statorique et de la fonction logique F_i

1.1. Représentation d'état du système

Pour une commande en tension de la machine asynchrone à cage, le modèle correspondant dans le repère lié au champ tournant est obtenu en considérant les composantes de tension (v_{ds} , v_{qs}) comme grandeurs de commande, et les variables (i_{ds} , i_{qs} , Φ_{dr} , Φ_{qr} , Ω) comme variables d'état. Ce modèle est régi par [Fu91] :

$$\dot{x} = f(x) + g(x) \cdot u \quad (1.17)$$

où :

$$x = (x_1, x_2, x_3, x_4, x_5)^t = (i_{ds}, i_{qs}, \Phi_{dr}, \Phi_{qr}, \Omega)^t$$

$$u = (u_1, u_2)^t = (v_{ds}, v_{qs})^t$$

$$f(x) = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \end{pmatrix} = \begin{pmatrix} -\gamma x_1 + \omega_s x_2 + \frac{k}{T_r} x_3 + p k x_4 x_5 \\ -\gamma x_2 - \omega_s x_1 + \frac{k}{T_r} x_4 - p k x_3 x_5 \\ \frac{L_m}{T_r} x_1 - \frac{1}{T_r} x_3 + (\omega_s - p x_5) \cdot x_4 \\ \frac{L_m}{T_r} x_2 - \frac{1}{T_r} x_4 - (\omega_s - p x_5) \cdot x_3 \\ \frac{pL_m}{JL_r} (x_3 x_2 - x_1 x_4) - \frac{C_r}{J} \end{pmatrix}$$

$$g(x) = (g_1(x) \quad g_2(x)) = \begin{pmatrix} \frac{1}{\sigma L_s} & 0 \\ 0 & \frac{1}{\sigma L_s} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

avec :

$$T_r = \frac{L_r}{R_r} ; \quad \sigma = 1 - \frac{L_m^2}{L_r L_s} ; \quad k = \frac{L_m}{L_r L_s - L_m^2} ; \quad \gamma = \frac{1}{\sigma L_s} \cdot \left(R_s - \frac{R_r L_m^2}{L_r^2} \right)$$

$\omega_{gl} = \omega_s - p x_5$: la vitesse du glissement.

En prenant les courants statoriques comme variables de commande, on obtient le modèle réduit de la machine comme suit :

$$\dot{x} = f(x) + g(x) \cdot u \quad (1.18)$$

avec :

$$x = (x_3, x_4, x_5)^t = (\Phi_{dr}, \Phi_{qr}, \Omega)^t$$

$$u = (u_1, u_2)^t = (i_{ds}, i_{qs})^t$$

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \end{bmatrix} = \begin{bmatrix} \frac{-x_3}{T_r} + (\omega_s - p \cdot x_5) \cdot x_4 \\ \frac{-x_4}{T_r} - (\omega_s - p \cdot x_5) \cdot x_3 \\ -\frac{C_r}{J} \end{bmatrix}$$

$$g(x) = [g_1(x) \quad g_2(x)] = \begin{bmatrix} \frac{L_m}{T_r} & 0 \\ 0 & \frac{L_m}{T_r} \\ -\frac{pL_m}{JL_r} x_4 & \frac{pL_m}{JL_r} x_3 \end{bmatrix}$$

Le modèle dans le référentiel fixe (α, β) est obtenu en prenant $\omega_s = 0$:

$$\dot{x} = f(x) + g(x) \cdot u \quad (1.19)$$

avec :

$$x = (x_3, x_4, x_5)^t = (\Phi_{\alpha r}, \Phi_{\beta r}, \Omega)^t ;$$

$$u = (u_1, u_2)^t = (v_{\alpha s}, v_{\beta s})^t$$

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \end{bmatrix} = \begin{bmatrix} \frac{-x_3}{T_r} - p \cdot x_4 \cdot x_5 \\ \frac{-x_4}{T_r} + p \cdot x_3 \cdot x_5 \\ \frac{C_r}{J} \end{bmatrix}$$

$$g(x) = [g_1(x) \quad g_2(x)] = \begin{bmatrix} \frac{L_m}{T_r} & 0 \\ 0 & \frac{L_m}{T_r} \\ -\frac{pL_m}{JL_r} x_4 & \frac{pL_m}{JL_r} x_3 \end{bmatrix}$$

L'avantage du modèle statorique par rapport à celui du champ tournant réside dans le fait qu'il ne nécessite pas le passage par une transformation dans un référentiel tournant, ce qui réduit considérablement les calculs. En plus, il est utilisé dans la conception des observateurs du flux et du couple.

1.8. Représentation du système sous forme canonique

Dans ce qui suivra, nous adopterons le modèle réduit du moteur asynchrone, régi par le système d'équations (1.19). Nous remarquerons que ce système est non-linéaire et qu'il existe un couplage étroit entre les différentes variables. Afin de mettre le système sous une forme canonique (ou normalisée), une transformation non-linéaire $\phi(x)$ est nécessaire. Cette transformation doit être difféomorphique, c'est-à-dire, différentiable, inversible et son inverse différentiable [Vid89][Isi89]. Pour cela, nous procédons en quatre étapes :

Étape 1 : Choix des grandeurs de sortie

Afin d'atteindre l'objectif de notre commande qui consiste en la régulation des deux grandeurs à savoir le flux et la vitesse, on choisit les variables de sorties suivantes [Kaz96]:

$$\begin{aligned} z_1 = h_1(x) &= \Phi_{\alpha r}^2 + \Phi_{\beta r}^2 = x_3^2 + x_4^2 \\ z_2 = h_2(x) &= \Omega = x_5 \end{aligned} \quad (1.20)$$

Étape 2 : Calcul du degré relatif

On calcule le degré relatif r_i associé à chaque grandeur de sortie z_i choisie, lequel degré correspond au nombre de fois qu'il faut dériver cette sortie pour faire apparaître explicitement une des grandeurs de commande.

– Sortie z_1 :

$$\dot{z}_1 = L_f h_1(x) + L_{g_1} h_1(x) \cdot u_1 + L_{g_2} h_1(x) \cdot u_2$$

$$\text{avec : } L_f h_1(x) = -\frac{2}{T_r} (x_3^2 + x_4^2), \quad L_{g_1} h_1(x) = \frac{2 L_m}{T_r} \cdot x_1, \quad L_{g_2} h_1(x) = \frac{2 L_m}{T_r} \cdot x_2$$

Le degré relatif associé à la grandeur de sortie z_1 est : $r_1 = 1$

– Sortie z_2 :

$$\dot{z}_2 = L_f h_2(x) + L_{g_1} h_2(x) \cdot u_1 + L_{g_2} h_2(x) \cdot u_2$$

$$\text{avec : } L_f h_2(x) = -\frac{C_r}{J}, \quad L_{g_1} h_2(x) = -\frac{p L_m}{J L_r} \cdot x_2, \quad L_{g_2} h_2(x) = \frac{p L_m}{J L_r} \cdot x_1$$

Le degré relatif associé à la grandeur de sortie z_2 est : $r_2 = 1$

La somme des degrés relatifs correspondants aux sorties $(x_3^2 + x_4^2)$ et x_5 est donnée par :

$$r_1 + r_2 = 1 + 1 = 2$$

Ceci nous permet de remarquer qu'il existe une dynamique interne non contrôlable, qui doit être stable (au moins bornée) [Isi89], et que l'ordre du nouveau système à commander est inférieur à celui du système initial.

La représentation normalisée de ce système nécessite un choix approprié de la transformation non-linéaire de coordonnées.

Étape 3 : Choix de la transformation $\phi(x)$

On opte pour la transformation non-linéaire de coordonnées suivante :

$$\begin{aligned} z_1 &= \phi_1(x) = x_3^2 + x_4^2 \\ z_2 &= \phi_2(x) = x_5 \\ z_3 &= \phi_3(x) = \text{atan}(x_4/x_3) + k \pi \end{aligned} \quad (1.21)$$

avec :

$$k = \begin{cases} 0 & \text{si } x_3 \geq 0 \\ 1 & \text{si } x_3 < 0 \end{cases}$$

La valeur k est utilisée pour assurer l'inversion de la transformation (1.21).

La transformation inverse est définie comme suit :

$$\begin{aligned} x_3 &= \sqrt{z_1} \cos z_3 \\ x_4 &= \sqrt{z_1} \sin z_3 \\ x_5 &= z_2 \end{aligned} \quad (1.22)$$

Et en se basant sur les équations (1.19) et (1.21), le système dans le nouvel espace d'état devient :

$$\begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \end{pmatrix} = \begin{pmatrix} -\frac{2}{T_r} z_1 \\ -\frac{C_r}{J} \end{pmatrix} + \begin{pmatrix} 2\frac{L_m}{T_r} x_3 & 2\frac{L_m}{T_r} x_4 \\ -\frac{pL_m}{JL_r} x_4 & \frac{pL_m}{JL_r} x_3 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad (1.23a)$$

$$\dot{z}_3 = p \cdot z_2 + \frac{L_m}{T_r} \left(\frac{x_3 u_2 - x_4 u_1}{z_1} \right) \quad (1.23b)$$

Le système (1.23a) peut être mis sous la forme matricielle suivante :

$$\dot{z}_r = f(z) + G(z) \cdot u \quad (1.24)$$

avec : $z_r = (z_1, z_2)^t$, $z = (z_1, z_2, z_3)^t$ et $G(z)$ étant la matrice de découplage.

Étape 4 : Retour d'état non-linéaire

Pour que le retour d'état non-linéaire puisse être défini, il est impératif que la matrice de découplage soit inversible. Ce qui signifie que son déterminant n'est pas nul.

Donc :

$$\det(G(z)) = \frac{2L_m}{T_r} \cdot \frac{pL_m}{JL_r} \cdot z_1 \neq 0 \quad (1.25)$$

En choisissant le retour d'état sous la forme :

$$u = G^{-1}(z) (v - f(z)) \quad (1.26)$$

nous obtenons les relations entre les nouvelles variables de commande v_1 et v_2 , et les variables u_1 et u_2 , définies comme suit :

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \frac{1}{\det(G(z))} \begin{pmatrix} \frac{pL_m}{JL_r} x_3 & -\frac{2L_m}{T_r} x_4 \\ \frac{pL_m}{JL_r} x_4 & \frac{2L_m}{T_r} x_3 \end{pmatrix} \begin{pmatrix} v_1 + \frac{2}{T_r} z_1 \\ v_2 + \frac{C_r}{J} \end{pmatrix} \quad (1.27)$$

Le nouveau système d'état peut être représenté par deux intégrateurs d'ordre 1 donnant deux sous-systèmes linéaires et entièrement découplés :

$$\begin{aligned} \dot{z}_1 &= v_1 \\ \dot{z}_2 &= v_2 \end{aligned} \quad (1.28)$$

Cependant, nous remarquons dans ce cas que le système d'équations (1.27) est plutôt complexe, et en plus il dépend étroitement des variables d'état, des paramètres de la machine et des perturbations externes. Cela le rend sensible à toute variation ou mal-identification, et donc sa représentation par un modèle approximatif n'est pas évidente.

Afin de minimiser la dépendance de ce système par rapport aux variations paramétriques et aux perturbations externes, tout en maintenant le découplage entre les deux sous-systèmes commandés par les variables de commandes v_1 et v_2 , nous proposons la relation suivante entre les commandes u et v :

$$u = G^{-1}(z) v' \quad (1.29)$$

De manière plus explicite, nous pouvons écrire :

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \frac{1}{\det(G(z))} \begin{pmatrix} \frac{pL_m}{JL_r} x_3 & -\frac{2L_m}{T_r} x_4 \\ \frac{pL_m}{JL_r} x_4 & \frac{2L_m}{T_r} x_3 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \quad (1.30)$$

Par conséquent, le système d'état global devient :

$$\begin{aligned} \dot{z}_1 &= -\frac{2}{T_r} \cdot z_1 + v_1' \\ \dot{z}_2 &= -\frac{C_r}{J} + v_2' \end{aligned} \quad (1.31)$$

Nous remarquons qu'il peut y avoir une autre écriture encore plus simple, et indépendante des paramètres de la machine, définie comme suit :

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} x_3 & -x_4 \\ x_4 & x_3 \end{pmatrix} \cdot \begin{pmatrix} v_1'' \\ v_2'' \end{pmatrix} \quad (1.32)$$

En considérant v_1'' et v_2'' comme les nouvelles variables de commande, la forme canonique correspondante au système régi par les équations (1.23a) est donnée par :

$$\begin{aligned} \dot{z}_1 &= -\frac{2}{T_r} z_1 + \frac{2L_m}{T_r} z_1 \cdot v_1'' \\ \dot{z}_2 &= -\frac{C_r}{J} + \frac{pL_m}{JL_r} z_1 \cdot v_2'' \end{aligned} \quad (1.33)$$

Le système résultant comporte un retour d'état défini par la boucle interne (1.32) et par un système non-linéaire (1.33), constitué de deux sous-systèmes mis sous forme canonique et dépendants chacun d'une seule commande v_i'' ($i = 1, 2$). La commande liée à un tel modèle est dite *commande décentralisée*.

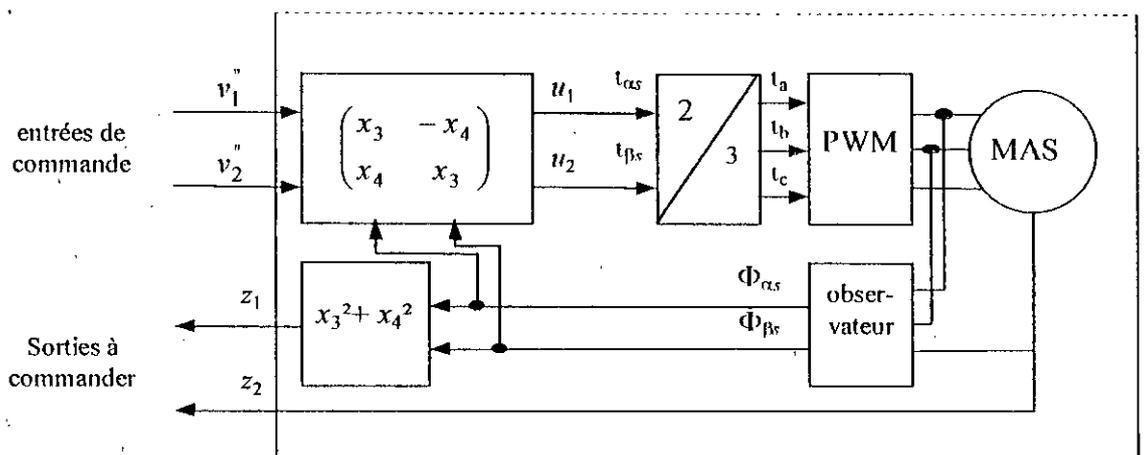


Figure 1.5 : Représentation du modèle du moteur asynchrone basé sur le retour d'état non-linéaire adopté

Le bloc décrivant la relation entre les commandes u et v , illustré à la figure 1.5, ne peut être remplacé par un système approximatif, car aucune incertitude entre u et v n'est tolérée. Ce qui signifie qu'une bonne observation des deux composantes du flux rotorique est nécessaire [Tad99a][Tad99b].

Dans notre approche, nous allons approximer, par une représentation floue, le modèle du système dynamique non-linéaire et incertain régi par les équations (1.33). Pour cela, on utilise la technique de modélisation floue des systèmes incertains introduite par Ben Ghalia [Gha94][Gha96a]. Une commande floue robuste basée sur ce modèle flou [Gha96b] va être ensuite appliquée.

1.9. Conclusion

Dans ce chapitre, nous avons présenté la modélisation analytique de la machine asynchrone. Sous certaines hypothèses simplificatrices, le modèle triphasé est réduit à un modèle équivalent biphasé à l'aide de la transformation de Park. Suivant le choix des variables d'état et de commande, le modèle est mis sous une représentation d'état dans les référentiels liés au stator et au champ tournant. Celui-ci est ensuite mis sous une forme canonique utilisant un retour d'état, permettant la commande décentralisée de la machine asynchrone et l'obtention d'un modèle mieux adapté à l'introduction des systèmes approximatifs.

Dans ce qui va suivre, nous optons pour le modèle réduit de la machine asynchrone. Les entrées de commande sont les composantes du courant statorique, et il convient d'utiliser un onduleur commandé par la technique de modulation de largeurs d'impulsions basée sur la stratégie d'hystérésis.

Chapitre II

Différentes Stratégies de Commande d'un Moteur Asynchrone

II.1. Introduction

Les applications industrielles, telles que la robotique, les servo-moteurs et autres, nécessitent des performances dynamiques et statiques très élevées ; c.-à-d. une réponse du couple très rapide. C'est pourquoi, la machine à courant continu reste jusqu'à nos jours très utilisée à cause de la simplicité de commande qu'elle possède. Elle constitue un optimum du point de vue dynamique que l'on essaye de retrouver avec les autres machines, en particulier la machine asynchrone, vu les divers avantages qu'elle présente. Néanmoins, cette dernière a un inconvénient majeur qui réside dans le couplage étroit entre les variables d'entrée, internes et de sortie. Par conséquent, il est nécessaire de lui adjoindre une commande adéquate. En effet, au début, la variation de vitesse se faisait soit à fréquence fixe et tension d'amplitude variable, soit en intervenant sur les caractéristiques électriques du rotor en utilisant un rhéostat externe, chose qui cause des pertes ohmiques considérables.

Par la suite, le développement de l'électronique de puissance et de la micro-électronique a permis l'amélioration des sources d'alimentation à fréquences variables. Le premier type de commande consistait à garder le rapport tension sur fréquence fixe [Cha84][Yu98]. L'invariance de ce dernier implique un couple de décrochage et un flux magnétique au régime permanent constants (Figure 2.1). Néanmoins, pour les faibles fréquences d'alimentation, les réactances sont faibles, et la chute de tension ohmique au niveau du stator provoque une diminution du flux statorique totalisé (et rotorique par la suite). Par conséquent, le couple de décrochage peut être même insuffisant pour faire démarrer le moteur. Les performances dynamiques restent cependant en dessous de celles obtenues avec la machine à courant continu [Cha84].

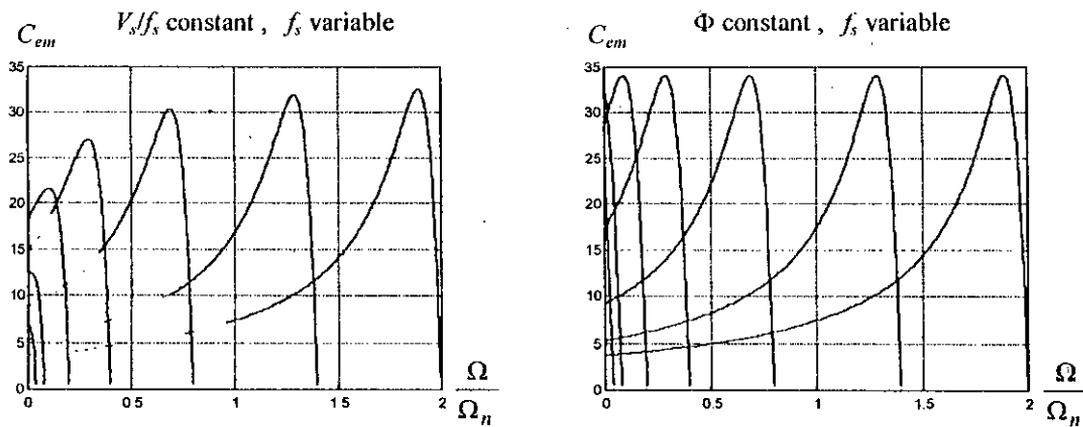


Figure 2.1 : Caractéristiques mécaniques pour différentes valeurs de la fréquence d'alimentation. a) avec V_s/f_s constant, b) avec flux constant

Depuis, dans le but d'améliorer les performances de la commande du moteur asynchrone et de minimiser le problème de couplage, d'autres stratégies ont vu le jour.

Dans ce chapitre, nous présenterons la commande par flux orienté (CFO) qui rend le moteur semblable, du point de vue commande, au moteur à courant continu. Nous introduirons ensuite le retour d'état linéarisant, qui permet le découplage entre le flux et le couple même en régime transitoire, en association avec la commande par placement de pôles. Nous adopterons la commande robuste à structure variable, vu la sensibilité aux incertitudes paramétriques et aux perturbations externes des deux commandes précédentes. Enfin, nous présenterons les concepts du réglage par logique floue et en vue de son application dans la commande en vitesse du moteur asynchrone.

II.2. Commande à flux orienté (CFO)

D'après le modèle (1.11), nous pouvons remarquer que le couple électromagnétique C_{em} est le résultat d'une interaction complexe entre les courants imposés dans les enroulements statoriques et les courants induits dans le rotor en court-circuit. Ainsi, toute variation du couple par variation des courants statoriques se traduit par une évolution du flux induit dans le rotor. Par conséquent, pour obtenir un contrôle dynamique performant du couple, il est nécessaire de réaliser un découplage des grandeurs *Couple-Flux* par l'adjonction d'une commande adéquate appelée *commande vectorielle*, ou *commande par flux orienté* (CFO) [Fu91][Bar93][Trz94].

Cette méthode de commande a été proposée pour la première fois en Allemagne en 1971 par Blaschke [Bla71], qui a introduit la méthode directe. Hasse a introduit ensuite la méthode indirecte. Le principe de cette technique consiste à ramener les performances de la machine asynchrone vers celles de la machine à courant continu à excitation séparée, où le courant inducteur contrôle le flux et le courant induit contrôle le couple, et cela en orientant le vecteur flux (rotorique, statorique ou d'entrefer) sur l'axe d dans un repère lié au champ tournant, ce qui signifie élimination de la composante en quadrature du flux et

qui se traduit par l'orthogonalité des deux vecteurs : courant statorique et flux rotorique, permettant ainsi une production optimale du couple.

Considérons l'orientation du flux rotorique. En substituant les conditions de la commande vectorielle $\Phi_{qr} = 0$ et $\Phi_{dr} = \Phi_r$ dans les équations du modèle (1.17), on aboutit au système d'équation suivant :

$$\begin{aligned} \frac{d}{dt} i_{ds} &= \frac{1}{L_1} \left(-\frac{\alpha}{T_r} \Phi_r + \frac{L_1}{T_1} i_{ds} - L_1 \omega_s i_{qs} + v_{ds} \right) \\ \frac{d}{dt} i_{qs} &= \frac{1}{L_1} \left(\alpha \omega_m \Phi_r + L_1 \omega_s i_{ds} + \frac{L_1}{T_1} i_{qs} + v_{qs} \right) \\ T_r \frac{d}{dt} \Phi_r + \Phi_r &= L_m i_{ds} \\ C_e &= \frac{p \cdot L_m}{L_r} \cdot \Phi_r \cdot i_{qs} \\ \omega_s &= \omega_m + \frac{L_m}{T_r} \cdot \frac{i_{qs}}{\Phi_{dr}} = \omega_m + \omega_{gl} \end{aligned} \quad (2.1)$$

D'après le système (2.1), il est clair que seule la composante directe du courant statorique i_{ds} commande le flux rotorique Φ_r , alors que le couple ne dépend que de la composante en quadrature i_{qs} , si le flux rotorique est maintenu constant. Par contre, les équations des tensions v_{ds} et v_{qs} montrent un couplage indésirable entre les diverses variables du moteur; ce qui nécessite l'utilisation d'un circuit de découplage lors de l'implémentation de la commande.

On prend le couple C_e et le flux Φ_r comme références, et les courants statoriques i_{ds} , i_{qs} , et la vitesse de glissement ω_{gl} comme sorties de commande, afin de *forcer* le moteur à suivre les deux références (Figure 2.2). Dans le cas de la commande en tension, les tensions statoriques v_{ds} , v_{qs} , et la vitesse de glissement ω_{gl} sont prises comme sortie de commande.

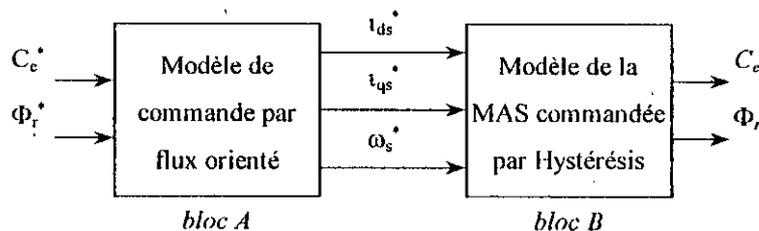


Figure 2.2. Schéma-bloc de la commande indirecte par orientation du flux rotorique.

Donc, le modèle de la commande par flux orienté (CFO) n'est autre qu'une inversion du modèle de la machine en respectant l'orientation du flux rotorique. L'orientation du flux est assurée par :

$$\omega_s = p x_5 + \frac{L_m}{T_r} \cdot \frac{u_2}{x_3} \quad (2.2a)$$

Dans le cas de la commande en courant, les variables de commande générées par la CFO sont données par :

$$u_1 = v_{ds} = \frac{1}{L_m} \left(T_r \frac{d\Phi_r^*}{dt} + \Phi_r^* \right) \quad (2.2b)$$

$$u_2 = v_{qs} = \frac{L_r}{pL_m} \cdot \frac{C_e^*}{\Phi_r^*}$$

avec C_e^* , Φ_r^* : le couple et le flux de référence.

Dans le cas où une régulation de vitesse est de rigueur, il suffit de reprendre le principe de la commande indirecte présenté dans le paragraphe précédent, et d'y adjoindre d'une part un régulateur de type PI pour la boucle de vitesse, et d'autre part un contrôle du flux rotorique, par réaction à travers un bloc de défluxage défini par la non-linéarité suivante :

$$\Phi_r^* = \begin{cases} \Phi_{nom} & \text{si } |\Omega| \leq \Omega_{nom} \\ \Phi_{nom} \cdot \frac{\Omega_{nom}}{\Omega} & \text{si } |\Omega| > \Omega_{nom} \end{cases} \quad (2.2c)$$

La référence du flux Φ_r^* est obtenue à partir d'un bloc de défluxage, et la variable de commande du couple C_e^* est générée par le régulateur de vitesse.

Le couple est borné de façon symétrique pour ne pas dépasser le couple maximal admissible par le moteur, ainsi que le courant maximal que peut supporter le convertisseur.

Le schéma de commande de cette structure est donné par la figure 2.3.

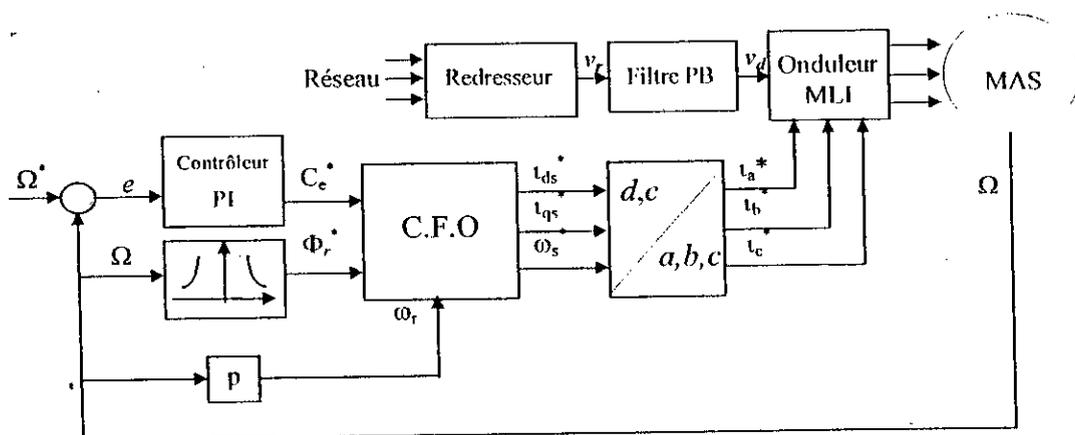


Figure 2.3 : Schéma de commande indirecte par flux orienté du moteur asynchrone commandé en courant

II.2.1. Résultats de simulation

Dans ce travail, la machine asynchrone utilisée possède les caractéristiques présentées en annexe A.

La dynamique de la commande indirecte est simulée pour les conditions suivantes :

- démarrage à vide pour une consigne de vitesse de 100 rad/sec
- application brusque d'un couple de charge de 10 Nm à l'instant $t = 0.5$ sec
- inversion de la consigne de vitesse de 100 rad/sec à -100 rad/sec appliquée à $t = 1.5$ sec

Les paramètres du régulateur PI sont calculés en fonction du point de fonctionnement, comme décrits dans [Bar93]. Pour un flux de référence de 1 Wb, les constantes sont : $K_p = 0.861$, $K_i = 0.005$ pour $C_r = 0$ et $K_p = 0.861$, $K_i = 0.10$ pour $C_r = 10$ Nm.

Les résultats de simulations obtenus sont illustrés par la figure 2.4.

D'après les réponses obtenues, on remarquera que la vitesse suit sa valeur de référence sans dépassement ni erreur statique à l'issue d'un temps de réponse d'environ 0.25 sec.

La commande vectorielle, quant à elle, est mise en évidence par l'annulation de la composante en quadrature du flux rotorique, ce qui se traduit par l'orientation du flux suivant l'axe d .

On constate également que l'effet produit par la brusque variation du couple est négligeable sur les deux composantes du flux. Ce résultat est prévisible vu que la commande est en courant et l'alimentation en tension.

L'inversion de vitesse provoque un pic du couple électromagnétique, alors que les deux composantes du flux rotorique restent constantes et continuent à suivre leurs références.

Il est important de noter que l'allure du courant i_{qs} est similaire à celle du couple électromagnétique, ce qui signifie qu'effectivement pour une référence du flux constant, le couple est contrôlé par le courant i_{qs} , ce qui est similaire au fonctionnement dynamique d'une machine à courant continu à excitation séparée.

Ces résultats de simulation nous permettent de conclure qu'un découplage effectif est obtenu entre les deux grandeurs : couple et flux de la machine. Ce qui signifie que l'objectif de la commande est atteint et que l'on peut contrôler séparément ces deux grandeurs.

Test de robustesse

Comme la commande vectorielle est connue pour sa sensibilité vis-à-vis des incertitudes et des variations paramétriques, en particulier la résistance rotorique, deux essais ont été effectués :

- Le premier consiste à faire varier la résistance rotorique par rapport à la résistance nominale R_{r0} de 90% à 150%, alors que les inductances (L_r, L_m, L_s) sont supposées connues avec certitude.
- Pour le second essai, les mêmes variations de la résistance rotorique sont maintenues ; quant aux inductances, elles sont supposées connues avec une incertitude de 5%.

Les résultats de simulation obtenues sont respectivement représentés par les figures 2.5 et 2.6. Nous remarquons que dans les deux cas une détérioration des performances dynamiques est d'autant plus notable que la variation de R_r augmente, ce qui se traduit par une perte totale du contrôle vectoriel ($\Phi_{qr} \neq 0$). Cette dégradation des réponses est plus accentuée dans le second essai.

La stratégie de commande indirecte par flux orienté présente l'inconvénient d'être très sensible aux variations des paramètres du moteur, surtout lorsqu'il s'agit de la variation de la constante de temps rotorique T_r . Souvent, nous rencontrons des difficultés pour identifier ces paramètres qui varient avec l'état de fonctionnement, à cause de phénomènes divers tels que l'effet de peau, la variation de la température ou la saturation magnétique. L'erreur entre les paramètres utilisés dans le modèle de commande et les paramètres réels se répercute sur les performances dynamiques obtenues en entraînant une mauvaise orientation du flux rotorique sur le repère choisi.

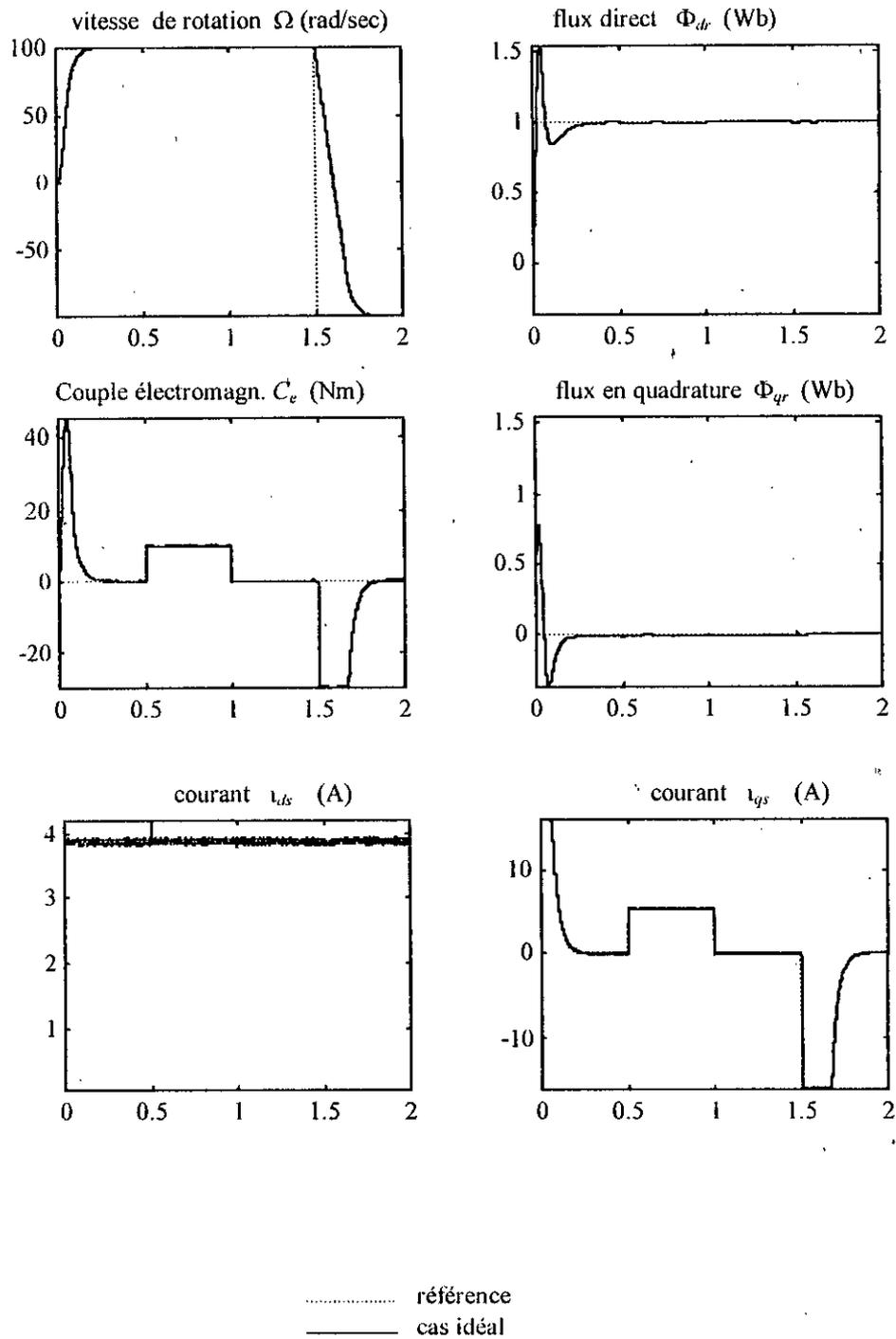


Figure 2.4 : Commande par flux orienté. Comportement dynamique de la machine dans le cas idéal.

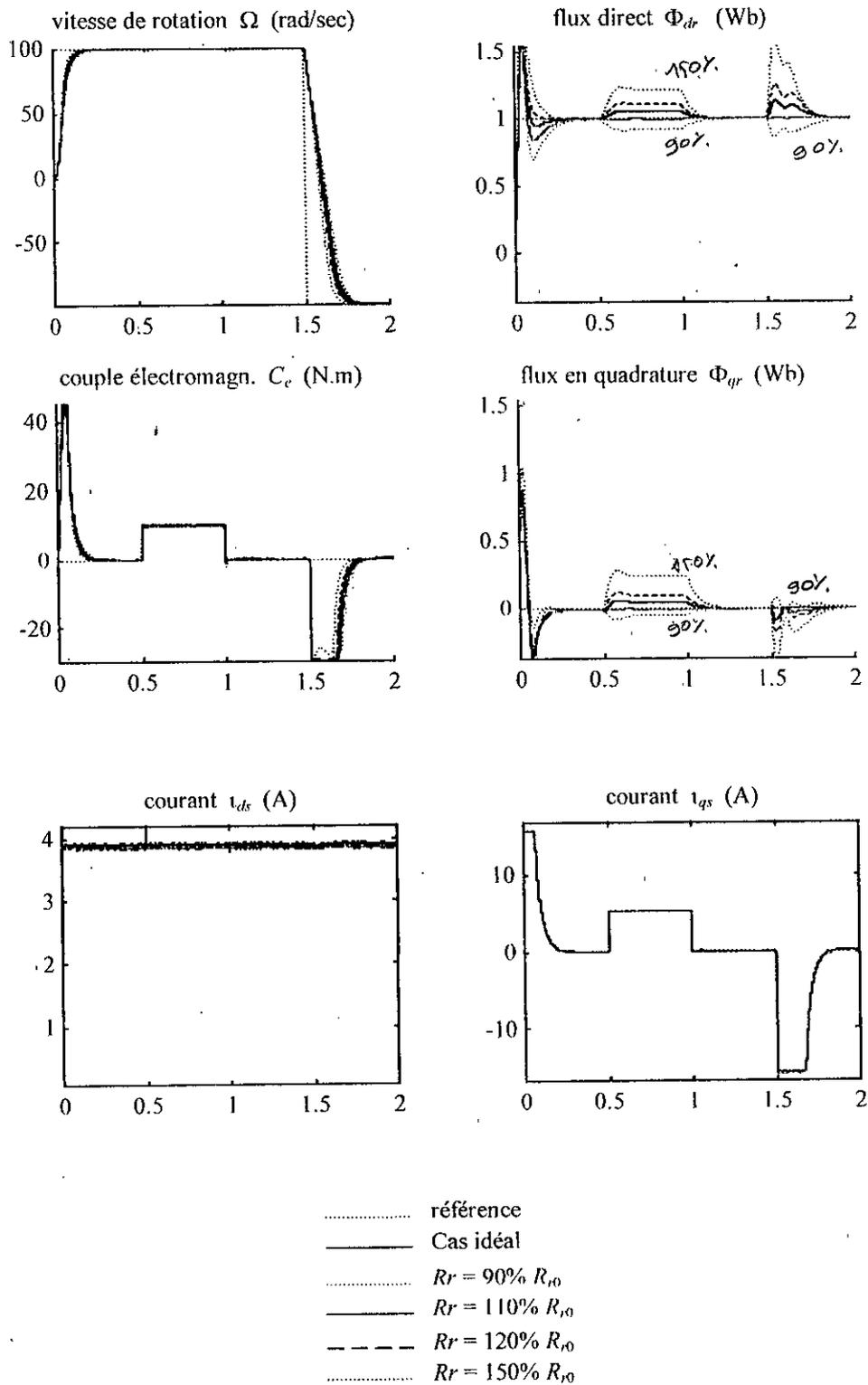


Figure 2.5 : Commande par flux orienté. Comportement de la machine lors d'une variation de R_r de 90% à 150% (sans variation des inductances).

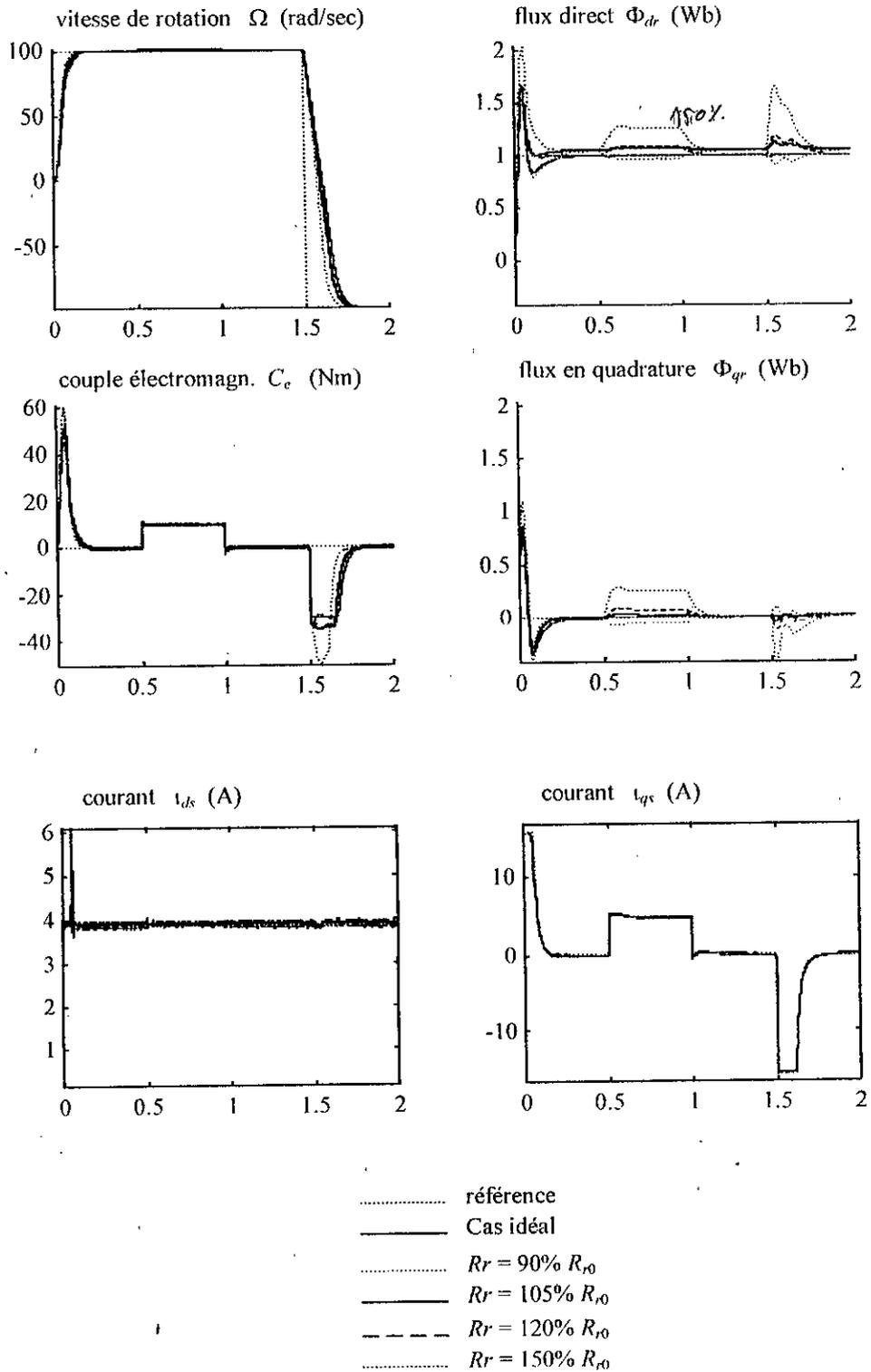


Figure 2.6 : Commande par flux orienté. Comportement de la machine lors d'une variation de R_r de 90% à 150% (avec une variation de 5% sur les inductances)

II.3. Commande par placement de pôles basée sur un retour d'état linéarisant

L'application du retour d'état non-linéaire, définie par l'équation (1.27), sur le modèle réduit du moteur asynchrone dans le repère (α, β) conduit à une linéarisation partielle [La199][Nib94]. Le schéma de commande correspondant est illustré par la figure 2.7.

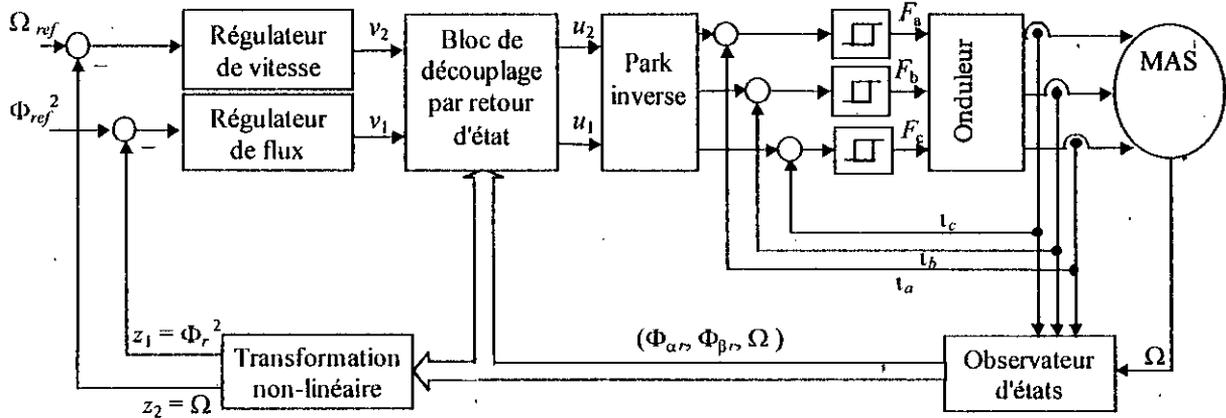


Figure 2.7 : Schéma de commande du moteur asynchrone basée un retour d'état linéarisant

La régulation de la vitesse et du flux se fait par un retour d'état linéarisant fournissant les deux variables de commande u_1 et u_2 . Les nouvelles variables de commande v_1 et v_2 , quant à elles, sont générées par les deux régulateurs de la boucle externe.

Une technique fréquemment utilisée pour le réglage du système linéaire ainsi obtenu est la technique dite par placement de pôles régie par les relations suivantes [La199][Bar00] :

$$\begin{aligned}
 v_1 &= k_{p1} \cdot (z_{1ref} - z_1) + k_{i1} \cdot \int_0^t (z_{1ref} - z_1) \cdot d\tau + \dot{z}_{1ref} \\
 v_2 &= k_{p2} \cdot (z_{2ref} - z_2) + k_{i2} \cdot \int_0^t (z_{2ref} - z_2) \cdot d\tau + \dot{z}_{2ref}
 \end{aligned}
 \tag{2.3}$$

avec z_{1ref}, z_{2ref} : le flux de référence et la vitesse de référence respectivement.

$$e_i = z_{iref} - z_i \quad (i = 1,2)$$

Les coefficients $k_{p1}, k_{i2}, k_{p3}, k_{i4}$ sont choisis de sorte que les polynômes $k_{p1} + k_{i2}s + s^2$ et $k_{p3} + k_{i4}s + s^2$ justifient le critère de stabilité d'Hurwitz.

II.3.1. Résultats de simulation

Dans les mêmes conditions que la section précédente, une simulation numérique de la commande par placement de pôle basée sur un retour d'état linéarisant a été effectuée (Figures 2.8 à 2.11).

Les résultats de simulation montrent le comportement dynamique de la machine lors du réglage de la vitesse. On note, dans tous les cas, que la variation du couple de charge n'affecte pas le flux en régime permanent, ce qui nous permet d'affirmer que le découplage entre les deux grandeurs couple et flux rotorique est bien réalisé.

Comme la commande par placement de pôles est connue pour sa sensibilité vis-à-vis des incertitudes et variations paramétriques, en particulier la résistance rotorique, et pour montrer qu'effectivement cette commande est réellement sensible à la variation de ce paramètre, l'essai suivant a été effectué. Celui-ci consiste à faire varier la résistance rotorique par rapport à la résistance nominale R_{r0} de 50% à 150%. L'erreur sur les inductances, qui sont supposées connues à 5% près, n'a pas beaucoup d'influence sur la dynamique du flux. (Figure 2.9)

Nous remarquons qu'une détérioration des performances dynamiques de la commande du flux rotorique est d'autant plus notable que la variation de R_r augmente. Par contre, la vitesse et le couple ne sont pas très affectés par les incertitudes paramétriques inhérentes à la machine.

La figure 2.10 montre la simulation du réglage dans les mêmes conditions d'essai que précédemment avec une charge inconnue par les régulateurs. Les performances de la commande de vitesse, dans ce cas, sont à leur tour détériorées. Pour éclaircir la dépendance de la vitesse par rapport au couple résistant, une simulation a été faite pour différents erreurs lors de l'estimation de C_r . Les résultats obtenus par ce test sont donnés par la figure 2.11.

Toutefois, cette technique de réglage s'avère très sensible à toute variation des paramètres internes de la machine et des perturbations externes qui peuvent survenir lors du fonctionnement de la machine.

Afin de pallier à ce problème, d'autres techniques plus robustes ont vu le jour, parmi lesquelles nous citons la commande par mode de glissement. Les avantages de ce type de commande sont importants, et très recherchés dans la commande des processus, à savoir : une grande précision, une bonne stabilité, et plus particulièrement une meilleure robustesse. Le paragraphe suivant est consacré à l'introduction de la commande à structure variable et les modes glissants associés au modèle présenté sous forme canonique (1.33).

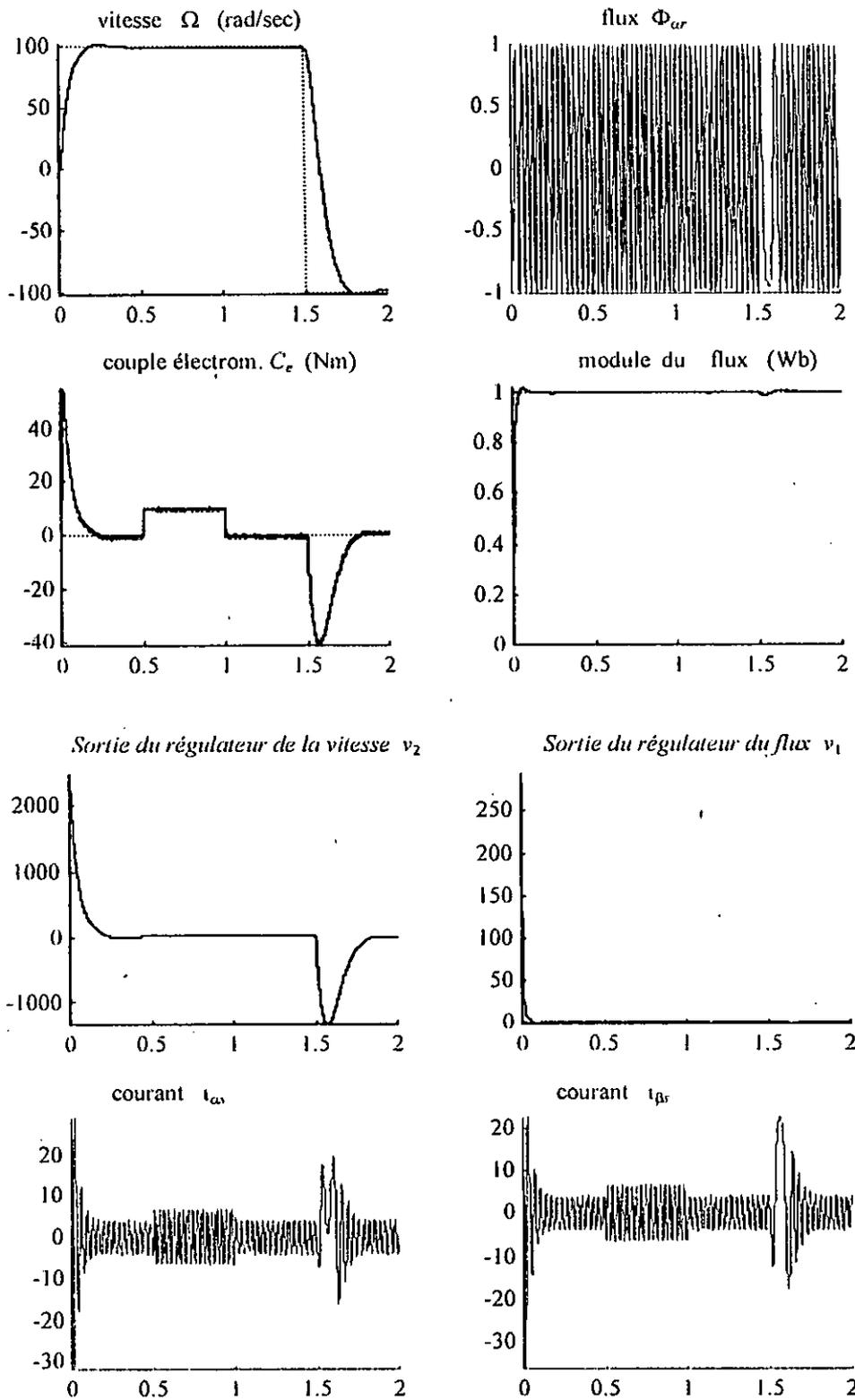
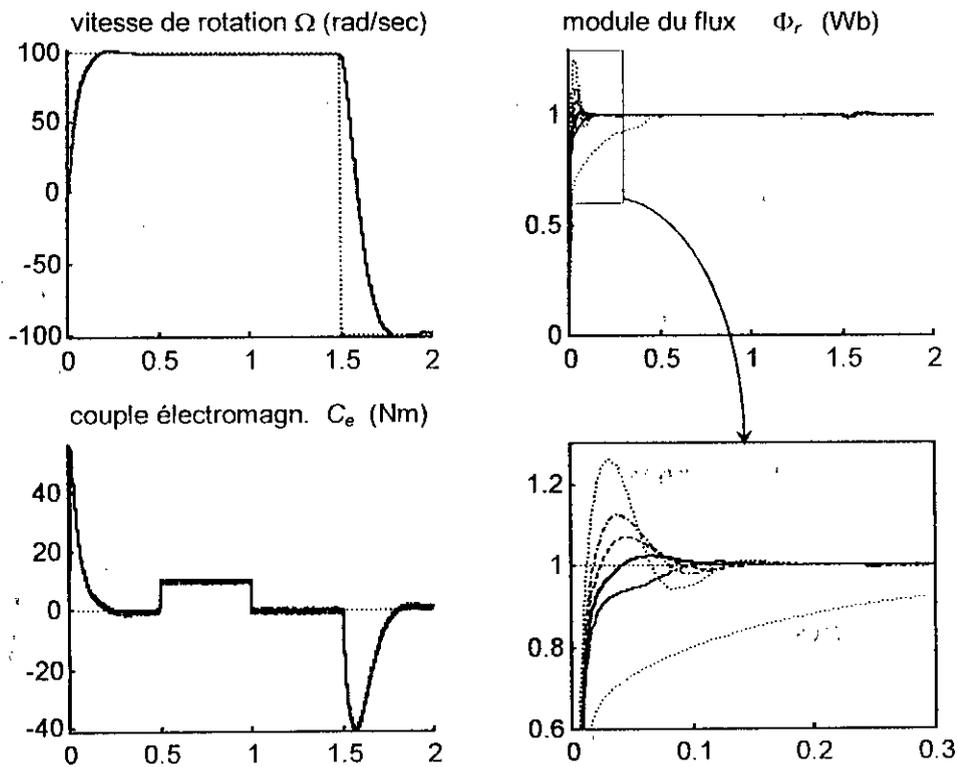
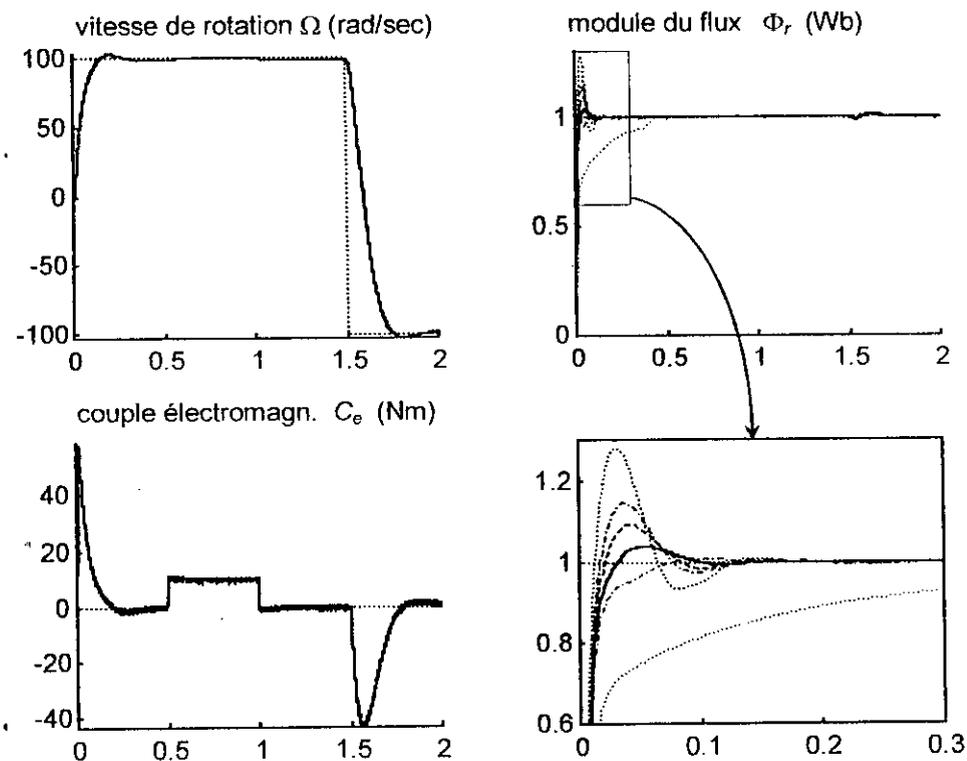


Figure 2.8 : Commande par placement de pôles.
Comportement dynamique du moteur dans le cas idéal.



a) Pas de variation sur les inductances



a) Avec une variation de 5% sur les inductances

..... référence ——— $R_r = 90\%R_{r0}$ - - - $R_r = 110\%R_{r0}$ $R_r = 150\%R_{r0}$
 $R_r = 50\%R_{r0}$ ——— Cas idéal - - - $R_r = 120\%R_{r0}$

Figure 2.9 : Commande par placement de pôles. Résultats de simulation pour différentes variations de R_r , avec estimation de la charge

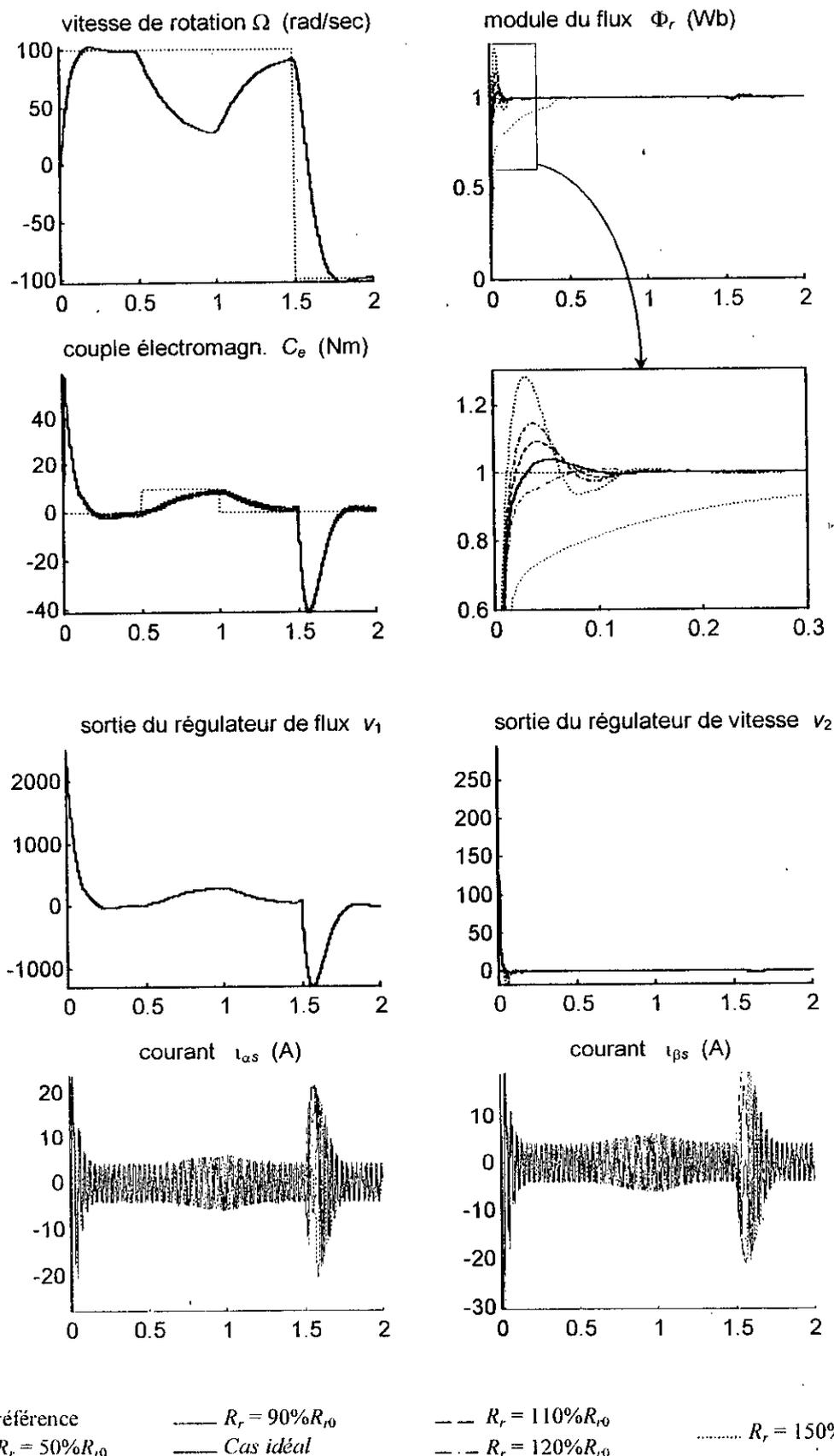


Figure 2.10 : Commande par placement de pôles. Résultats de simulation pour différentes variations de R_r (sans estimation de la charge et avec variation de 5% sur les inductances)

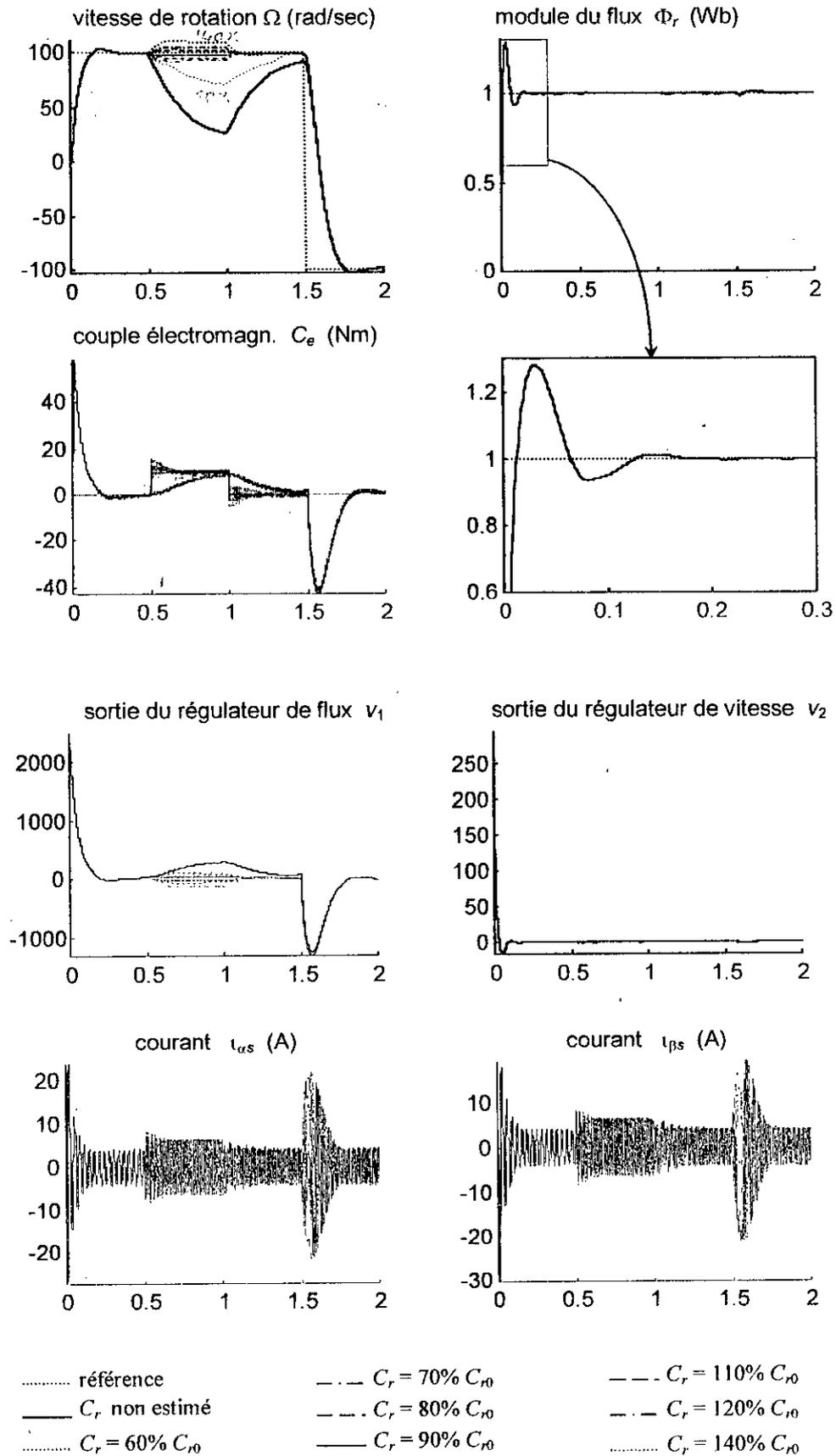


Figure 2.11 : Commande par placement de pôles. Résultats de simulation pour différentes erreurs sur l'estimation de C_r (avec $R_r = 150\% R_{r0}$ et une variation de 5% sur les inductances)

II.4. Commande à structure variable

La commande à structures variables (CSV) a été développée dans les années soixante par l'équipe du professeur Emelyanov [Eme62] à partir des études théoriques du mathématicien Fillipov [Fil60], elle est basée sur les équations différentielles à second membre discontinu. Cette commande constitue une classe de systèmes de commande non-linéaires, dont la structure change en fonction de l'état du système (Figure 2.12). Cependant, le système régi par une des structures n'est pas nécessairement stable, la combinaison de ces structures crée un mode appelé mode de glissement [Dec88][Utk92][You99][Slo91]. C'est-à-dire, la trajectoire d'état du système est amenée vers une hypersurface dite surface de glissement (dont la dimension est généralement inférieure au nombre d'états du système) et commute autour de cette surface jusqu'au point d'équilibre. Une commande à structure variable avec des modes glissants est souvent dite Commande par modes de glissement (CMG).

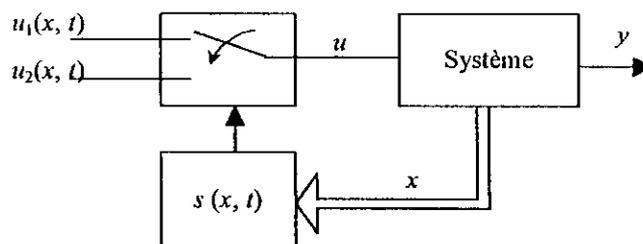


Figure 2.12 : Aspect général d'une commande à structure variable.

II.4.1. Conception des régulateurs à structures variables

Considérons un système multi-entrées / multi-sorties (MIMO) décrit par la forme matricielle suivante :

$$\dot{x} = f(x) + g(x) \cdot u \quad (2.4)$$

où :

$$x = \begin{pmatrix} x_1 \\ \dots \\ x_n \end{pmatrix} ; f(x) = \begin{pmatrix} f_1(x) \\ \dots \\ f_n(x) \end{pmatrix} ; g(x) = \begin{pmatrix} g_{11}(x) & \dots & g_{m1}(x) \\ \dots & & \dots \\ g_{1n}(x) & \dots & g_{mn}(x) \end{pmatrix} ; u = \begin{pmatrix} u_1(x) \\ \dots \\ u_m(x) \end{pmatrix}$$

La mise en œuvre des contrôleurs par mode de glissement d'un tel système se fait en trois étapes, à savoir :

- choix de la surface de glissement

- ▣ établissement des conditions d'existence des modes glissants et de convergence vers la surface de glissement
- ▣ détermination des lois de commande.

II.4.1.1. Choix de la surface de glissement

D'après le système d'équations (2.5), nous pouvons remarquer qu'il peut exister m surfaces de glissement relatives à chaque commande u_i . Leur choix doit être fait de manière à ce que leur intersection, qui représente en réalité la surface de glissement S , permette d'avoir le comportement désiré du système ainsi qu'une bonne poursuite de cette surface S .

La surface de glissement S peut être définie par:

$$S = \bigcap_{i=1}^m S_i \quad (2.5)$$

$$S_i = \{ x \in \mathbb{R}^n / s_i(x) = 0 \} \quad ; \quad i = 1, \dots, m$$

avec :

$$s(x) = \begin{pmatrix} s_1(x) \\ \dots \\ s_m(x) \end{pmatrix} \quad (2.6)$$

S_i : surface de glissement correspondante à l'entrée u_i

$s_i(x)$: fonction de commutation

$s(x)$: vecteur des fonctions de commutation

Le degré relatif de chaque surface $s_i(x)$ doit être égal à 1, c.-à-d. la commande u_i doit apparaître dès la première dérivée de $s_i(x)$.

II.4.1.2. Choix des grandeurs de commande

En ce qui concerne la conception de la loi de commande à structure variable u_i ($i = 1, \dots, m$), elle est donnée par :

$$u_i(x, t) = \begin{cases} u_i^+(x, t) & \text{si } s_i(x) > 0 \\ u_i^-(x, t) & \text{si } s_i(x) < 0 \end{cases} \quad (2.7)$$

Le rôle de ces lois de commande est de faire en sorte que chaque vecteur d'état x , se trouvant en dehors de la surface de glissement régie par la condition $s(x) = 0$, est attiré vers cette même surface, en un temps fini. Une fois la surface atteinte, le mode de glissement a

lieu, et les dynamiques désirées du système sont effectives. De ce fait, le système devient insensible aux incertitudes paramétriques et aux perturbations externes.

II.4.1.3. Condition d'existence du mode de glissement

Cependant, il est important de déterminer la condition avec laquelle l'état x atteint et glisse sur la surface. Cette condition est dite *condition d'atteignabilité*. Pour satisfaire cette condition, une fonction de Lyapunov est utilisée ; elle est régie par :

$$V(x,t) = \frac{1}{2} s'(x) \cdot s(x) \quad (2.8)$$

La condition d'atteignabilité de la surface de glissement S est :

$$\dot{V}(x,t) = [s(x)]^t \cdot \dot{s}(x) = \sum_{i=1}^m s_i(x) \cdot \dot{s}_i(x) < 0, \quad \text{pour tout } x \in \mathbb{R}^n - S \quad (2.9)$$

En se basant sur cette condition, deux principales techniques de commande ont été élaborées, à savoir la technique de Filippov [Fil60], et celle de la commande équivalente proposée par V.I.Utkin [Utk93]. Dans ce qui va suivre, nous nous intéressons uniquement à cette dernière.

II.4.1.4. La commande équivalente

Dans ce cas, la loi de commande u est définie comme :

$$u = u_{eq} + u_n \quad (2.10)$$

u_{eq} et u_n sont respectivement la commande équivalente et la commande d'atteignabilité.

Ces deux commandes peuvent être obtenues à partir du développement de la dérivée de la loi de commutation $\dot{s}(x,t)$ [You98].

$$\dot{s}(x,t) = \left(\frac{\partial s}{\partial x} \right)^t \cdot \dot{x} + \frac{\partial s}{\partial t} = \left(\frac{\partial s}{\partial x} \right)^t \cdot \{ f(x) + g(x) \cdot u \} + \frac{\partial s}{\partial t} \quad (2.11)$$

Ce qui nous permet d'écrire la loi de commande de la manière suivante :

$$u = \left\{ \left(\frac{\partial s}{\partial x} \right)^t \cdot g(x) \right\}^{-1} \cdot \left\{ \dot{s}(x,t) - \left(\frac{\partial s}{\partial x} \right)^t f(x) - \frac{\partial s}{\partial t} \right\} \quad (2.12)$$

avec la condition d'existence :

$$\det \left\{ \left(\frac{\partial s}{\partial x} \right)^t \cdot g(x) \right\} \neq 0 \quad (2.13)$$

En se basant sur (2.12), la commande équivalente qui correspond au régime glissant idéal et qui vérifie la condition $\dot{s}(x,t) = 0$, sera donnée par :

$$u_{eq} = - \left\{ \left(\frac{\partial s}{\partial x} \right)' \cdot g(x) \right\}^{-1} \cdot \left\{ \left(\frac{\partial s}{\partial x} \right)' f(x) + \frac{\partial s}{\partial t} \right\} \quad (2.14)$$

Par conséquent, l'équation d'état du système en régime glissant est obtenue en remplaçant la commande u_{eq} dans le système d'équations (2.4) :

$$\dot{x} = \left\{ I - g(x) \cdot \left\{ \left(\frac{\partial s}{\partial x} \right)' \cdot g(x) \right\}^{-1} \cdot \left(\frac{\partial s}{\partial x} \right)' \right\} f(x) - g(x) \cdot \left\{ \left(\frac{\partial s}{\partial x} \right)' \cdot g(x) \right\}^{-1} \cdot \left(\frac{\partial s}{\partial x} \right)' \quad (2.15)$$

Puisqu'en pratique le système réel présente des incertitudes paramétriques, non paramétriques et structurelles, et comme la phase d'atteignabilité représente une partie importante dans les dynamiques transitoires du système, la commande équivalente se trouve insuffisante à forcer la variable d'état à atteindre la surface de glissement. Il est donc important de spécifier une loi de commande susceptible de permettre au système de suivre une dynamique désirée. Cette dernière doit satisfaire la condition d'atteignabilité (2.9) qui garantit l'existence du mode de glissement et qui correspond à la commande u_n de l'équation (2.10). La loi de commande, dans ce cas, peut être exprimée par :

$$u = \left\{ \left(\frac{\partial s}{\partial x} \right)' \cdot g(x) \right\}^{-1} \cdot \left\{ \dot{s}_d(x,t) - \left(\frac{\partial s}{\partial x} \right)' f(x) - \frac{\partial s}{\partial t} \right\} \quad (2.16)$$

avec $\dot{s}_d(x,t)$ la dynamique désirée de la fonction de commutation [Gha95b][Mor85].

$$u_n = \left\{ \left(\frac{\partial s}{\partial x} \right)' \cdot g(x) \right\}^{-1} \cdot \dot{s}_d(x,t)$$

Le plus souvent, la loi de commande discontinue relative à u_n est de type relais, elle est décrite par :

$$\dot{s}_d(x) = -K \cdot \text{sgn}(s(x)) \quad (2.17)$$

K : matrice diagonale des gains ; $K = \text{diag}[K_1, \dots, K_m]$

sgn : un champ de vecteurs contenant des fonctions signe

$$s(x) = \begin{pmatrix} s_1(x) \\ \dots \\ s_m(x) \end{pmatrix}$$

En spécifiant cette dynamique, nous pouvons prévoir la vitesse de convergence de l'état du système vers la surface. Pour chaque fonction de commutation $s_i(x)$, ($i = 1, \dots, m$), la vitesse de convergence ainsi que la robustesse de la commande à structure variable sont fortement liées au gain K_i [Gha95b][Dec88].

II.4.2. Cas des systèmes mis sous forme canonique

Le système étudié est mis sous forme canonique en blocs. Chaque bloc représente un sous système mono-entrée / mono-sortie défini par l'équation différentielle :

$$\dot{x}^i = f^i(x) + g^i(x) \cdot u_i(x) \quad ; \quad i = 1, \dots, m \quad (2.18)$$

avec :

$$x^i = \begin{pmatrix} x_1^i \\ \dots \\ x_{r_i}^i \end{pmatrix} ; \quad f^i(x) = \begin{pmatrix} f_1^i(x) \\ \dots \\ f_{r_i}^i(x) \end{pmatrix} = \begin{pmatrix} x_2^i \\ \dots \\ a^i(x) \end{pmatrix} ; \quad g^i(x) = \begin{pmatrix} g_1^i(x) \\ \dots \\ b^i(x) \end{pmatrix} = \begin{pmatrix} 0 \\ \dots \\ g_{r_i}^i(x) \end{pmatrix}$$

avec : $\sum_{i=1}^m r_i \leq n$

i : l'indice de l'entrée

r_i représente le degré relatif correspondant à l'entrée u_i

$x^i \in \mathbb{R}^{r_i}$;

$f^i(x) \in \mathbb{R}^{r_i}$; $g^i(x) \in \mathbb{R}^{r_i \times 1}$

$x \in \mathbb{R}^n$ le vecteur d'état ; n : étant la dimension de tout le système.

Généralement, les surfaces de glissement sont choisies comme étant des hyperplans pour des raisons de stabilité et de simplicité. J.J. Slotine propose une équation de la forme générale pour la détermination de la surface de glissement $s_i(x^i)$ assurant la convergence [Slo84][Slo91] :

$$s_i(x^i) = \lambda_1 e_1 + \lambda_2 e_2 + \dots + \lambda_{r_i} e_{r_i} \quad (2.19)$$

avec : $e_j = x_j^{*i} - x_j^i$; $j = 1, \dots, r_i$

$(x_1^{*i}, \dots, x_{r_i}^{*i})^t$ constitue la trajectoire de poursuite du $i^{ème}$ sous-système.

λ_j étant les coefficients de l'hyperplan.

L'équation (2.19) peut être écrite de la manière suivante :

$$s(x) = \lambda_1 e + \lambda_2 \dot{e} + \dots + \lambda_{r_i} \frac{d^{(r_i-1)} e}{dt} \quad (2.20)$$

Les coefficients λ_j sont choisis de telle manière que le polynôme :

$$\Gamma(p) = p^{r_i} + \lambda_{r_i} p^{(r_i-1)} + \dots + \lambda_1 \quad (2.21)$$

soit un polynôme de Hurwitz. Par conséquent, une fois que le système est dans la surface, sa dynamique est asymptotiquement stable et converge exponentiellement.

En dehors de la surface, la condition d'atteignabilité (2.22) doit être vérifiée pour chaque sous système :

$$s_i(x) \cdot \dot{s}_i(x) < 0 \quad ; \quad i = 1, \dots, m \quad (2.22)$$

En dérivant la surface $s(x)$ nous obtenons l'équation suivante :

$$\dot{s}_i(x^i) = \lambda_1 \dot{e}_1 + \lambda_2 \dot{e}_2 + \dots + \lambda_{r_i} \dot{e}_{r_i} \quad (2.23)$$

La substitution de la relation (2.18) dans l'équation (2.23) nous ramène à :

$$\dot{s}_i(x^i) = \lambda_1 e_2 + \lambda_2 e_3 + \dots + \lambda_{r_i-1} e_{r_i} + \dot{x}^{*i} - (a^i(x) + b^i(x) \cdot u_i) \quad (2.24)$$

avec $\lambda_{r_i} = 1$

Ce qui nous permet d'aboutir à l'équation définissant la loi de commande suivante :

$$u_i = (\lambda_1 e_2 + \lambda_2 e_3 + \dots + \lambda_{r_i-1} e_{r_i} + \dot{x}^{*i} - a^i(x) - \dot{s}_{id}(x^i)) \div b^i(x) \quad (2.25)$$

Dans ce cas, la dynamique désirée assurant la condition de stabilité (2.22) est choisie comme suit :

$$\dot{s}_{id}(x) = -K \cdot \text{sgn}(s_i(x^i)) \quad (2.26)$$

II.4.3. Inconvénients de la CSV

Malgré les avantages de la commande à structure variable, cette dernière souffre de deux problèmes majeurs. Le premier réside dans le fait que la propriété de l'insensibilité n'est présente que si le système est dans la surface de glissement. Le comportement des états qui sont en dehors de la surface devient sensible aux incertitudes paramétriques et aux perturbations externes, et la robustesse de la commande n'est vérifiée que dans une plage limitée d'incertitudes [Utk92][Utk93][Ish93]. Cependant, le système peut ne pas converger asymptotiquement, ou encore converger en un temps très grand, et la commande perdra ses avantages.

Le second problème concerne le *phénomène du chattering*. Dans la conception et l'analyse de la commande à structures variables, il est supposé que la fréquence de commutation entre les structures est infinie. Mais en pratique, cela est impossible pour plusieurs raisons, parmi lesquelles, nous pouvons citer l'existence du temps de retard lié au calcul de la loi de commande, et la limitation physique des organes de commande qui ne peuvent pas suivre la commutation théorique des deux structures. Comme conséquence de cet inconvénient, le système va commuter autour de la surface au lieu de glisser sur cette surface comme le montre la figure 2.13. Ce phénomène est indésirable, puisqu'il peut être une source d'excitation des dynamiques à hautes fréquences qui ne figurent pas dans la modélisation [Dec88][Utk92].

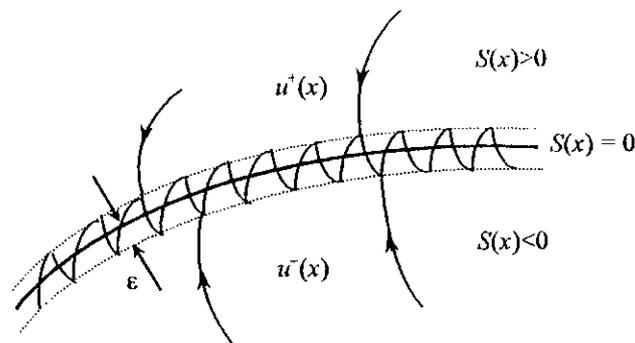


Figure 2.13 : Mise en évidence de l'écart de poursuite lors de l'implémentation pratique

Pour remédier au premier inconvénient, c.-à-d. pour réduire le temps d'atteignabilité, plusieurs approches ont été développées dans la littérature, à savoir :

- l'utilisation d'un contrôleur à grand gain suggéré dans [You78] (équation 2.26), et qui peut malheureusement conduire à la saturation et accentuer le phénomène de chattering.
- la condition de convergence (2.9) peut prendre la forme suivante [Slo91] :

$$[s(x)]^t \cdot \dot{s}(x) = \sum_{i=1}^m s_i(x) \cdot \dot{s}_i(x) \leq -\eta \cdot \|s(x)\| \quad (2.27)$$

avec η une constante positive, qui définira le temps de convergence vers la surface de glissement. La surface S sera atteinte en un temps t_r fini déterminé par [Slo91] :

$$t_r \leq \frac{s(t=0)}{\eta} \quad (2.28)$$

- une fonction de commutation variable dans le temps peut être éventuellement utilisée pour éliminer la phase d'atteignabilité où le vecteur d'erreur initial est supposé nul. Cette méthode est utilisée dans le cas d'une commande à structures variables avec un élément intégrateur pour les systèmes du premier ordre [Chu95a][Chu95b].

II.4.4. Réduction du phénomène de chattering

Afin de pallier au problème du chattering, nous avons retenu la solution qui consiste à choisir une commande discontinue qui varie en fonction de la distance entre la variable d'état et la surface de glissement [Dec88][Sab81][Sab89].

Pour une commande à structure variable du deuxième ordre, la commande discontinue peut être écrite sous la forme [Har86][Ham93] :

$$\Delta u = [\alpha_1 \cdot e_1 + \alpha_2 \cdot e_2 + k] \cdot \text{sgn}(s(x)) \quad (2.29)$$

Sur la base de ces travaux, nous pouvons généraliser l'expression de la commande discontinue pour les systèmes mis sous forme canonique, et aboutir à l'expression suivante :

$$\Delta u_i = \left[\sum_{j=1}^{r_i} (\alpha_j \cdot |e_j| + k) \right] \cdot \text{sgn}(s_i(x')) \quad ; \quad i = 1, \dots, m \quad (2.30)$$

avec α_k est la pente de la fonction $K(s_i)$ suivant la $j^{\text{ème}}$ direction du plan de la surface S_i de dimension r_i .

k est la valeur minimale de $K(s_i)$ qui assure la robustesse en mode de glissement, en tenant compte de la plage des incertitudes (Figure 2.14).

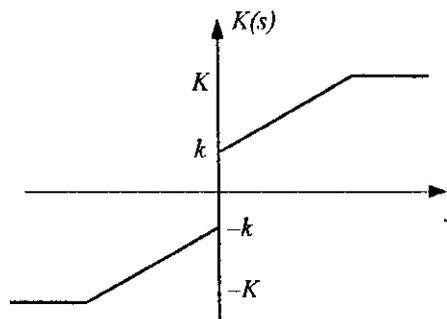


Figure 2.14 : Réduction du phénomène du chattering par une forme linéaire de la fonction discontinue

L'augmentation de k a pour effet une augmentation de la vitesse d'atteignabilité, mais cela présente l'inconvénient majeur d'induire un chattering important, c'est-à-dire un bruit ayant un spectre étendu et d'amplitude élevée. Ce qui peut nuire aux organes de commande, et conduire à la présence de parasites indésirables dans les grandeurs de sortie.

L'absence de commutation ($k = 0$), par contre, a l'avantage de réduire le phénomène, mais au prix de la robustesse de la commande.

D'une manière générale, la commande discontinue u_n est une fonction non-linéaire de la distance entre l'état du système et la surface de glissement, comme le montre la figure 2.15.

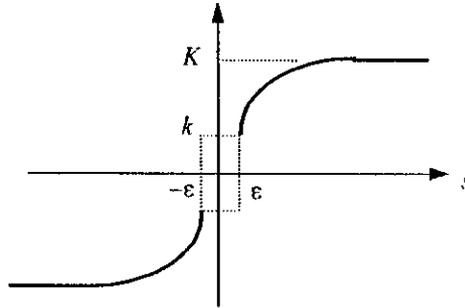


Figure 2.15 : Forme générale de la commande discontinue réduisant le chattering

K : correspond à la valeur maximale de l'organe de commande que peut supporter le système.

ε : doit être pris en compte dans l'implémentation pratique du processus car il influence directement sur la fréquence de commutation des semi-conducteurs et sur la poursuite de la trajectoire désirée. Un compromis est fait lors de la conception ; quand ε est très petit, la fréquence de commutation est très élevée et l'écart de poursuite autour de la surface est réduit, et vice versa [Utk92][Bar00].

II.4.5. Application au moteur asynchrone

L'application de la CSV dans la commande du moteur asynchrone a connu un intérêt remarquable [Utk93][Cha93][Pan94][Sob96a][Bar98][Utk99]. Dans notre cas, nous l'avons appliquée en tenant compte du modèle mis sous forme canonique, régi par les équations (2.1) et (2.5). Ceci peut être résumé par les étapes suivantes :

– *Choix de la surface de glissement* : Les fonctions de commutation des deux sous-systèmes sont de la forme :

$$\begin{aligned} s_1(z_1) &= \lambda_1 e_1(z_1) = \lambda_1 (\Phi_r^2 - \Phi_r^2) = \lambda_1 (z_{1ref} - z_1) \\ s_2(z_2) &= \lambda_2 e_2(z_2) = \lambda_2 (\Omega_{rref} - \Omega_r) = \lambda_2 (z_{2ref} - z_2) \end{aligned} \quad (2.31)$$

– *Choix des grandeurs de commande* : En utilisant l'équation (2.25), les deux lois de commande correspondant au réglage du flux et de la vitesse sont définies par :

$$v_1 = \frac{-\lambda_1 \cdot \dot{z}_{1ref} + \lambda_1 \cdot \frac{2}{T_r} z_1 - \dot{s}_{1d}}{\frac{2L_m}{T_r} z_1} \quad (2.32)$$

$$v_2 = \frac{-\lambda_2 \cdot \dot{z}_{2ref} + \lambda_2 \cdot \frac{C_r}{J} - \dot{s}_{2d}}{\frac{2L_m}{JL_r} z_1} \quad (2.33)$$

avec :

$$\dot{s}_{id}(x) = -K_i \cdot \text{sgn}(s_i(x^i)) \quad ; \quad i = 1, 2 \quad (2.34)$$

II.4.6: Résultats de simulations

Afin de mettre en évidence les avantages apportés par la commande à structure variable relativement à celle par placement de pôles, une simulation numérique a été effectuée dans les mêmes conditions d'essai que précédemment et avec une plage de variation de la résistance rotorique R_r de 50% à 200% par rapport à la résistance nominale.

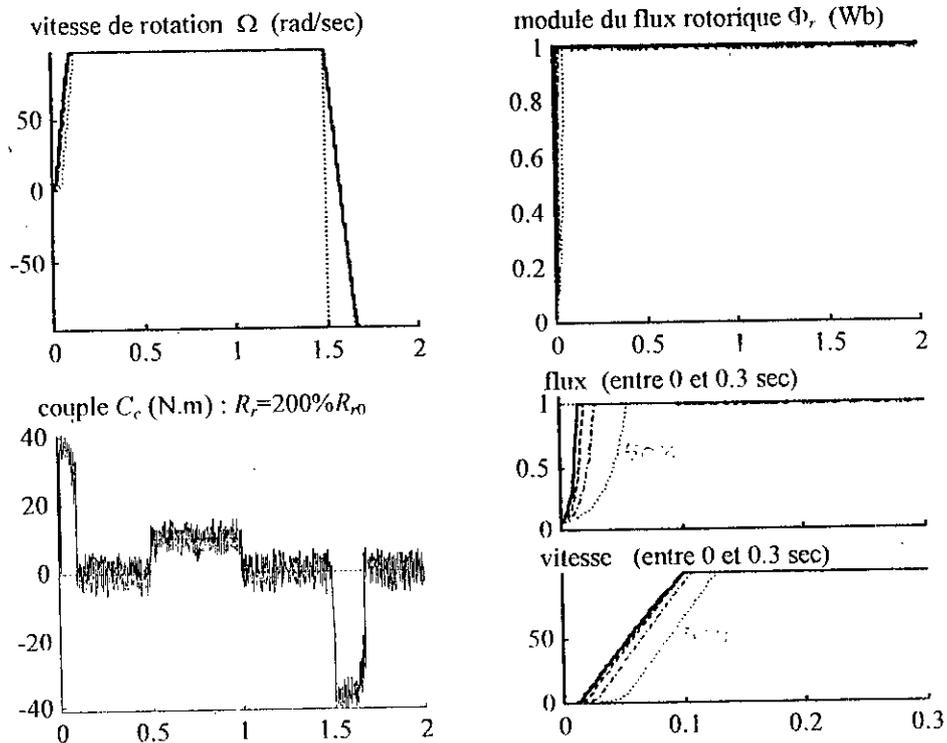
Les résultats de simulation montrent la dynamique de réglage de la vitesse et le rejet de toute perturbation. On note que, dans tous les cas, la variation du couple de charge n'affecte pas le flux en régime permanent, ce qui nous permet d'affirmer que le découplage entre les deux grandeurs, couple et flux rotorique, est réalisé.

Les résultats de simulation ainsi obtenus dans le cas de la commande à structure variable à relais sont représentés aux figures 2.16a et 2.16b. Les résultats montrent clairement une meilleure robustesse par rapport à la commande par placement de pôles vis-à-vis de la variation de la résistance rotorique R_r et de la perturbation externe. Néanmoins, des oscillations très prononcées apparaissent, et qui peuvent s'avérer néfastes pour les composants semi-conducteurs et le fonctionnement du moteur.

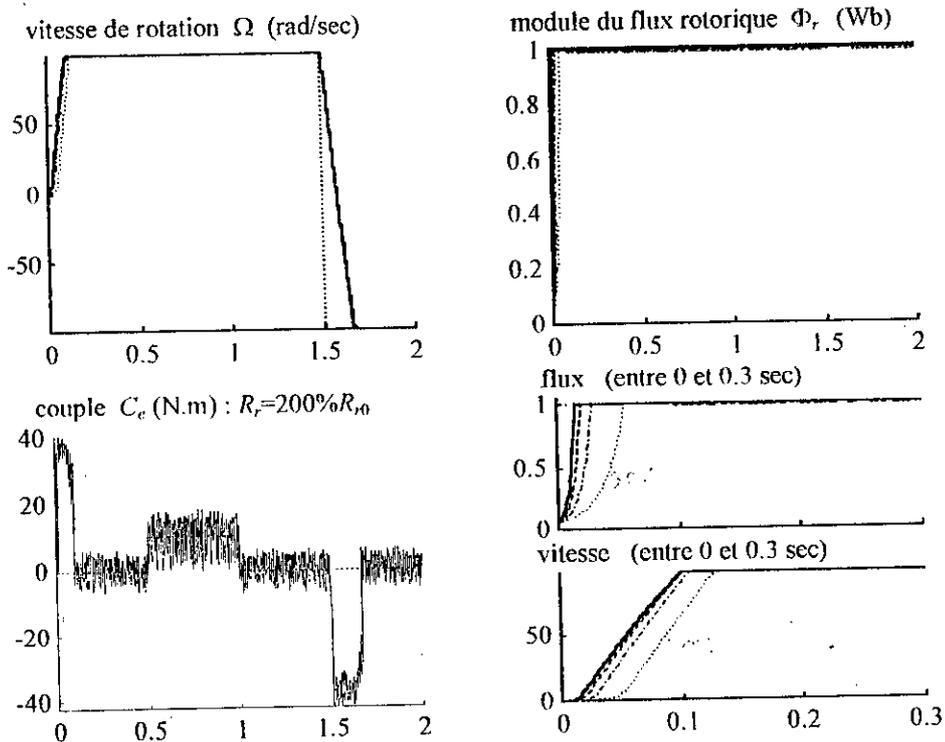
La figure 2.17 confirme que la commande à structure variable adoucie ($k = 0$) permet une réduction de ces oscillations, sauf que dans ce cas, on a une perte de la robustesse, vu la sensibilité de la vitesse par rapport au couple résistant.

On apprécie, sur la figure 2.18, les améliorations apportées par la commande à structures variables. Un choix judicieux des paramètres ε et k est nécessaire pour un meilleur fonctionnement. On note que le paramètre k influe sur la robustesse et le chattering, alors que ε agit sur la précision et la fréquence de commutation des semi-conducteurs.

L'étude de la robustesse montre que la commande par mode de glissement offre l'avantage d'être, par rapport à la commande par placement de pôles et la commande par flux orienté, insensible aux variations des paramètres internes de la machine et des perturbations externes, résultats prévisibles de par la nature de cette commande.



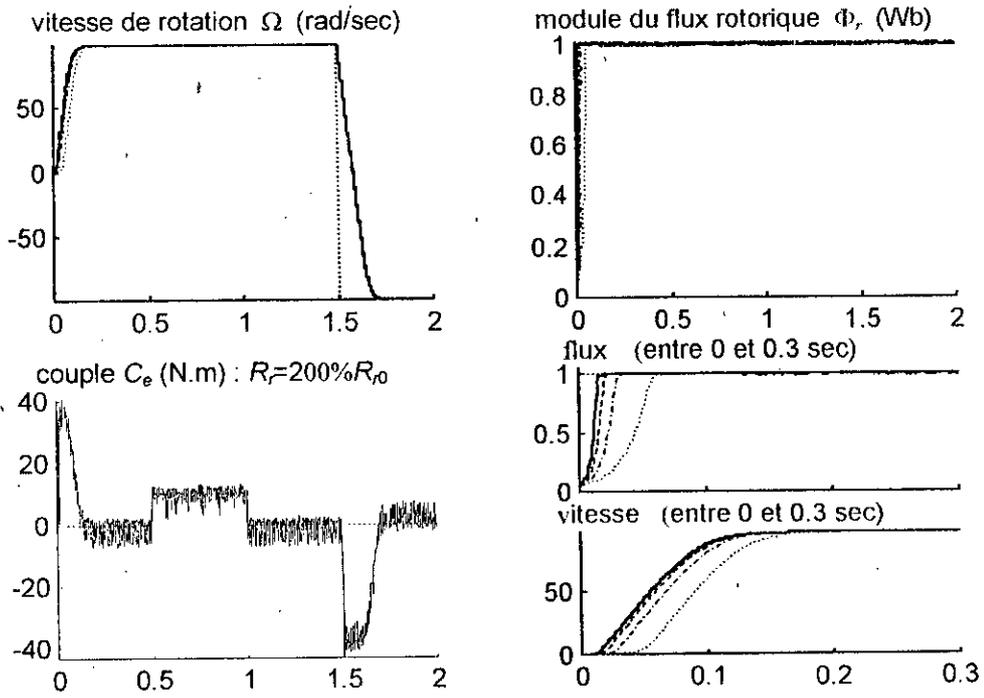
a) avec estimation de la charge



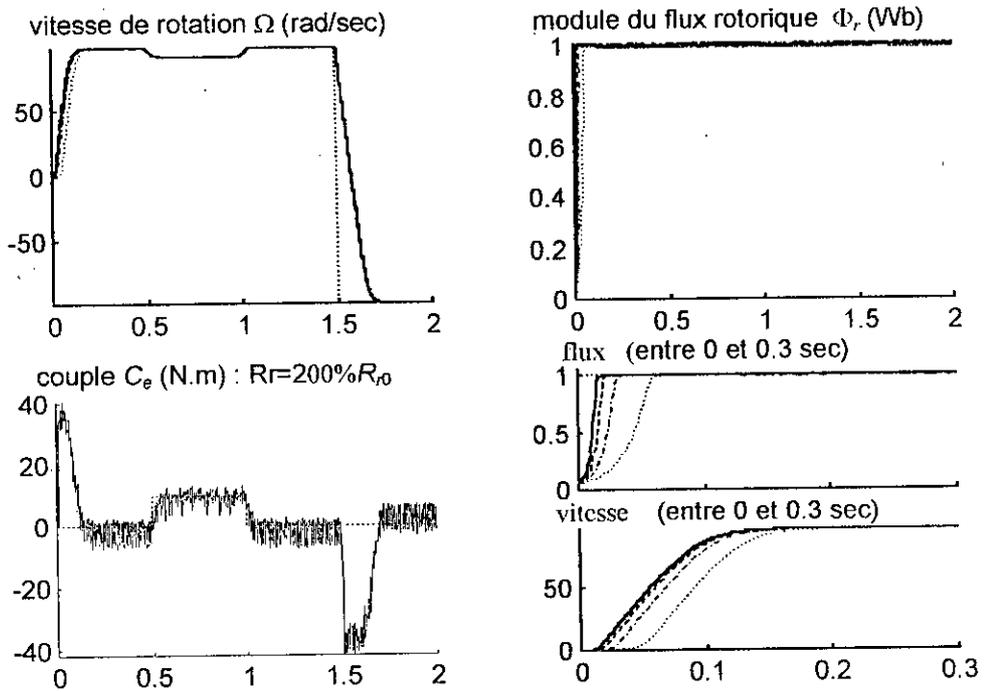
b) sans estimation de la charge

..... référence $R_r = 50\%R_{r0}$ - - - $R_r = 150\%R_{r0}$
 - - - $R_r = 100\%R_{r0}$ ——— $R_r = 200\%R_{r0}$

Figure 2.16 : Commande à structure variable à relais.
Résultats de simulation pour différentes variations de R_r .



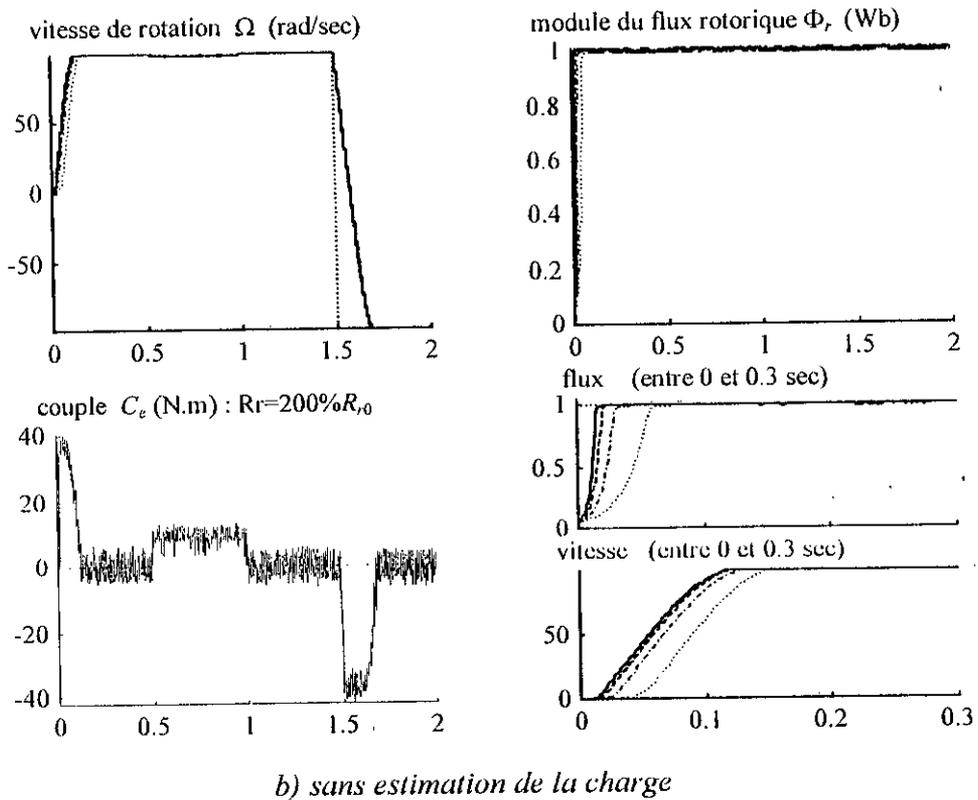
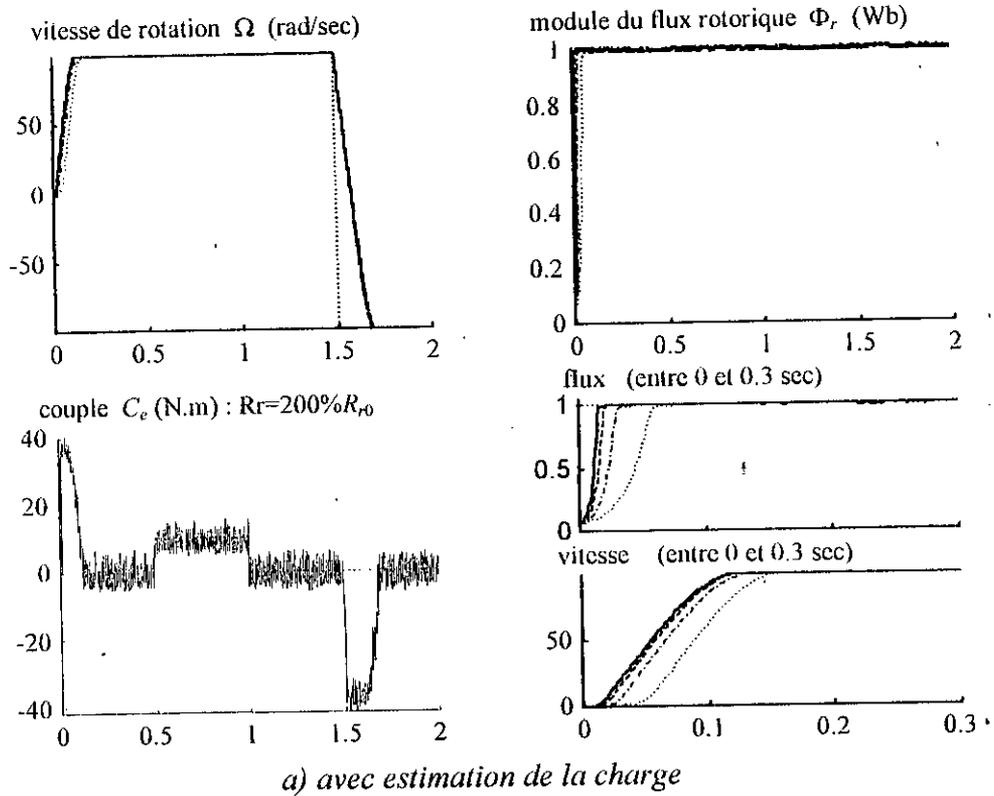
a) avec estimation de la charge



b) sans estimation de la charge

..... référence $R_r = 50\%R_{r0}$ - - - $R_r = 150\%R_{r0}$
 - - - $R_r = 100\%R_{r0}$ - - - $R_r = 200\%R_{r0}$

Figure 2.17 : Commande à structure variable adoucie ($k=0$).
 Résultats de simulation pour différentes variations de R_r .



..... référence $R_r = 50\%R_{r0}$ - - - $R_r = 150\%R_{r0}$
 - - - $R_r = 100\%R_{r0}$ ——— $R_r = 200\%R_{r0}$

Figure 2.18 : Commande à structure variable semi-adoucie ($k \neq 0$).
 Résultats de simulation pour différentes variations de R_r .

II.5. Commande par logique floue

II.5.1. Introduction

La logique floue a fait son apparition en 1965 avec les travaux de Lotfi Zadeh [Zad65]. Ce dernier a formalisé la représentation et le traitement de connaissances imprécises ou approximatives afin de traiter des systèmes de grande complexité ou mal connus. La logique floue intervient dans la manipulation des connaissances imparfaites et elle est survenue comme une alternative efficace pour de tels systèmes [Bou95].

Basées sur l'imitation des aspects approximatifs du raisonnement humain, les méthodes à base de règles linguistiques font intervenir la notion des ensembles flous et des citations conditionnelles linguistiques de type "Si - Alors" pour résoudre des problèmes de décision ou pour décrire le comportement d'un système inconnu ou incertain.

L'une des motivations de l'utilisation des ensembles flous est dans la modélisation de l'expertise humaine. Généralement, la perception de l'homme pour un système donné n'est pas basée sur des modèles mathématiques précis.

Dans cette partie, nous allons présenter les fondements théoriques de la logique floue et l'utilisation du raisonnement approximatif basé sur les expressions linguistiques en vue de la commande des systèmes dynamiques, entre autres la commande en vitesse du moteur asynchrone.

II.5.2. Théorie des ensembles flous

La notion des ensembles flous est basée sur le principe de l'appartenance partielle d'un élément x à un ensemble donné A . Ceci permet l'utilisation des limites mal définies de ces ensembles, de situation intermédiaire entre le tout et le rien (presque vrai), le passage progressif de l'appartenance d'un ensemble à un autre et l'utilisation de nombres approximatifs (environ 3) [Tle99][Men95]. Pour plus d'éclaircissement, nous présentons quelques principes de base de la théorie des ensembles flous.

Soit X l'univers de discours ou l'ensemble des valeurs possibles d'une variable x .

– Un ensemble classique A de X (Figure 2.19) est défini par une fonction caractéristique qui prend soit la valeur 0 ou 1, définie par :

$$\mu_A(x) : X \rightarrow \{0,1\} \quad (2.35)$$

avec $\mu_A(x) = 1$ si $x \in A$ et $\mu_A(x) = 0$ si $x \notin A$. L'appartenance d'un élément x à l'ensemble A est soit vraie (si $\mu_A(x) = 1$) soit fausse (si $\mu_A(x) = 0$).

– Un ensemble flou A est une généralisation d'un ensemble classique, défini par une fonction dite *fonction d'appartenance* $\mu_A(x)$, représentant le degré d'appartenance d'un élément réel x à l'ensemble flou A :

$$A = \{(x, \mu_A(x)) / x \in X\} ; \text{ avec : } \mu_A(x) : X \rightarrow [0,1] \quad (2.36)$$

On adopte souvent la notation suivante pour représenter un ensemble flou [Dri93][Men95] :

$$A = \int_X \mu_A(x) / x \text{ si } X \text{ est continu} \quad (2.37)$$

$$A = \sum_{x \in X} \mu_A(x) / x \text{ si } X \text{ est discret}$$

Selon l'exemple de la figure 2.19, la discontinuité paraît absurde : à un jour près, l'humain peut passer d'une catégorie à l'autre. L'attribut "jeune" est difficile à modéliser avec la logique classique booléenne. En outre, en logique floue, on introduit des fonctions d'appartenance qui définissent, en fonction de la variable "âge", les degrés d'appartenance à chaque état. Ces fonctions font progressivement passer l'individu d'un état à l'autre, ce qui est beaucoup plus naturel. On se rapproche donc du raisonnement humain.

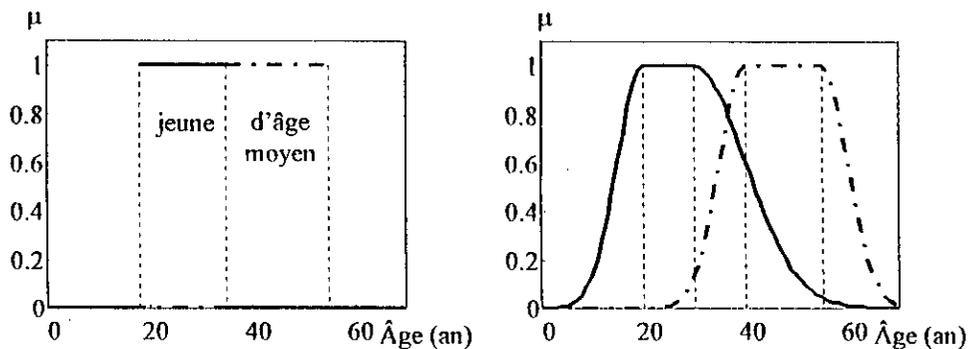


Figure 2.19 : Fonction d'appartenance. a) logique classique, b) logique floue

II.5.2.1. Quelques caractéristiques des ensembles flous

Le *support* de A est la partie de l'univers de discours X sur laquelle la fonction d'appartenance de A n'est pas nulle.

$$\text{supp}(A) = \{ x \in X / \mu_A(x) > 0 \} \quad (2.38)$$

Le *noyau* de A est l'ensemble des éléments de X pour lesquels la fonction d'appartenance de A vaut 1.

$$\text{noyau}(A) = \{ x \in X / \mu_A(x) = 1 \} \quad (2.39)$$

L'ensemble A est dit *ensemble flou normal* si son noyau est non vide. Il est considéré comme étant *convexe* si pour tout $x_1, x_2 \in X$ et $\lambda \in [0,1]$, on a :

$$\mu_A(\lambda \cdot x_1 + (1 - \lambda) \cdot x_2) \geq \min\{\mu_A(x_1), \mu_A(x_2)\} \quad (2.40)$$

Un ensemble flou dont le support est un point x_0 de X avec $\mu_A(x_0) = 1$ est dit *un singleton flou* centré sur x_0 .

II.5.2.2. Fonctions d'appartenance

Généralement, le choix de la forme des fonctions d'appartenance dépend plus de l'expert, que des justifications mathématiques [Kel97]. Il existe une panoplie de formes de fonctions d'appartenance qui peuvent être utilisées pour la conception d'un ensemble flou, par exemple :

- Les fonctions d'appartenance triangulaires et trapézoïdale (Figure 2.20a) qui sont les fonctions les plus simples et les plus utilisées.
- Les fonctions gaussienne, en forme de cloche généralisée, sigmoïdale (Figures 2.20b-d) qui sont souvent utilisées dans les problèmes tels que les systèmes neuro-flous nécessitant l'adaptation des paramètres de ces fonctions et requérant des fonctions d'appartenances différentiables.

Les expressions relatives aux fonctions d'appartenance et leurs représentations sont mentionnées dans l'annexe B1.

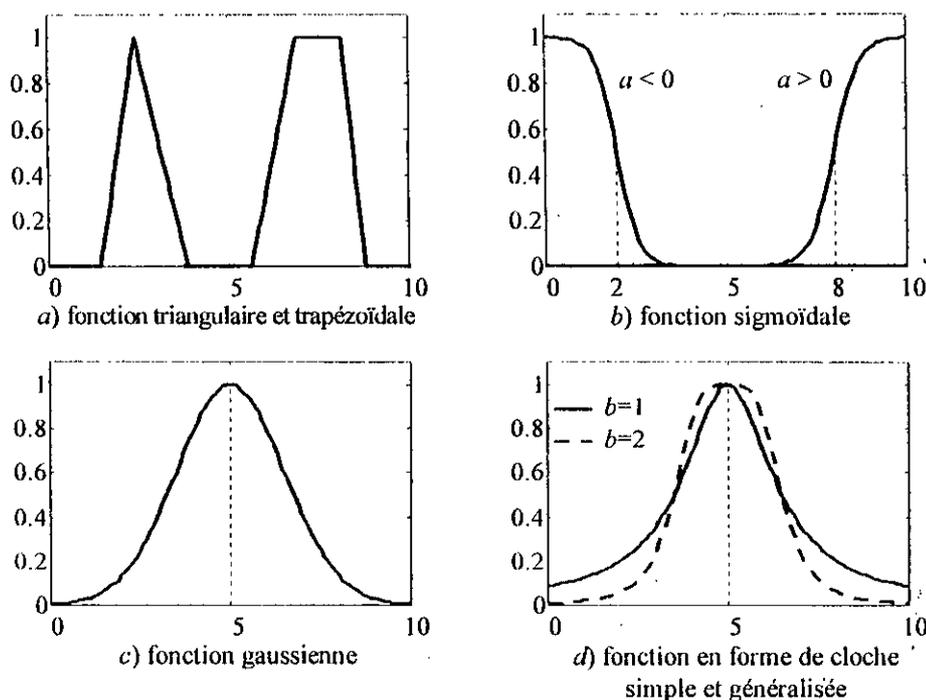


Figure 2.20 : Formes usuelles des fonctions d'appartenance

II.5.2.3. Opérations de base sur les ensembles flous

Par rapport aux opérations logiques classiques (ET, OU et Complément), la logique floue possède, quant à elle, des opérations similaires sur les ensembles flous, appelées : *Conjonction*, *Disjonction* et *Complément*, respectivement.

Soient A et B deux ensembles flous dans X définies par leurs fonctions d'appartenance respectives $\mu_A(x)$ et $\mu_B(x)$.

– Les degrés d'appartenance de l'intersection et de l'union des ensembles flous A et B sont représentés par des opérations de conjonction $T(\cdot, \cdot)$ et de disjonction $S(\cdot, \cdot)$ respectivement :

$$\mu_{A \cap B}(x) = T(\mu_A(x), \mu_B(x)) \quad (2.41)$$

$$\mu_{A \cup B}(x) = S(\mu_A(x), \mu_B(x))$$

$T(\cdot, \cdot)$ et $S(\cdot, \cdot)$ sont appelés des opérateurs de la norme triangulaire T -norme et T -conorme, respectivement.

– Le degré d'appartenance de \bar{A} (la *négation* de A) est spécifié par un opérateur de négation $N(\cdot)$ satisfaisant la condition :

$$N(0) = 1 \quad \text{et} \quad N(1) = 0 \quad (2.42)$$

$$N(a) \leq N(b) \quad \text{si} \quad a \geq b \quad (\text{monotonie})$$

Les opérateurs $T(\cdot, \cdot)$ et $S(\cdot, \cdot)$ doivent satisfaire certaines conditions et doivent vérifier la généralisation de la loi de De Morgan, données à l'annexe B2.

Des exemples des opérateurs triangulaires T -norme, de leurs conjoints T -conorme et de la négation sont regroupés dans un tableau présenté en annexe B3.

II.5.2.4. Produit cartésien des ensembles flous

Soient A_1, A_2, \dots, A_n des ensembles flous dans les univers de discours X_1, X_2, \dots, X_n respectivement. Le produit cartésien de $A_1 \times A_2 \times \dots \times A_n$ est un ensemble flou dans l'espace $X_1 \times X_2 \times \dots \times X_n$ ayant pour fonction d'appartenance la fonction multivariable suivante :

$$\mu_{A_1 \times \dots \times A_n}(x) = T(\mu_{A_1}(x_1), \mu_{A_2}(x_2), \dots, \mu_{A_n}(x_n)) \quad (2.43)$$

telle que $x = (x_1, \dots, x_n)$ un vecteur de $X_1 \times X_2 \times \dots \times X_n$.

T : opération de T -norme multivariable.

II.5.3. Variable linguistique

Le Professeur L.Zadeh en 1975 [Zad75] a annoncé : « En se retirant de la précision en face de la complexité du système, il est naturel d'explorer ce qu'on peut appeler les variables linguistiques, dont les valeurs ne sont plus des nombres mais des noms ou des phrases dans un langage artificiel ou naturel (...). La motivation de l'utilisation des mots ou des phrases au lieu des nombres, c'est que les caractérisations linguistiques en général sont moins spécifiques que les caractérisations numériques. »

Une variable linguistique est, par définition, le triplet $(V, X, T(V))$ dans lequel V est le nom de la variable, X l'univers de discours, et $T(V)$ l'ensemble de termes linguistiques (ou ensembles flous) dans X qui peuvent caractériser V .

Exemple: $V = \{ \text{Température} \}$; $X = [0^\circ \text{ } 100^\circ]$; $T(V) = \{ \text{très froide, froide, tiède, chaude} \}$.

II.5.4. Principe d'extension

D'une manière générale, le principe d'extension est un procédé susceptible de traiter des données vagues [Dri93][Bou95]. De ce fait, il permet d'étendre une application réelle, $\phi : X_1 \times \dots \times X_n \rightarrow Y$, à une application floue $\tilde{\phi} : P_f(X_1) \times \dots \times P_f(X_n) \rightarrow P_f(Y)$ en opérant sur les ensembles flous.

$P_f(X_i)$ représente une partition floue de X_i (ou la famille des ensembles flous engendrant l'univers de discours X_i).

La fonction d'appartenance de l'ensemble flou F_y résultant de l'extension de l'application ϕ sur les ensembles flous A_1, \dots, A_n est définie par :

$$\mu_{F_y}(y) = \sup_{\substack{(x_1, \dots, x_n) \in X_1 \times \dots \times X_n \\ \phi(x_1, \dots, x_n) = y}} \left\{ \min(\mu_{A_1}(x_1), \dots, \mu_{A_n}(x_n)) \right\} \quad (2.44)$$

telle que $\phi^{-1}(y)$ existe, et $\mu_{F_y}(y) \neq 0$ [Dri93][Bou95].

$$F_y \in P_f(Y) \quad ; \quad A_i \in P_f(X_i) \quad ; \quad i = 1, \dots, n$$

D'une manière générale, nous pouvons utiliser au lieu de la composition sup-min la composition (T-conorme-T-norme), pour aboutir à une généralisation de ce principe :

$$\mu_{F_y}(y) = \sup_{\substack{(x_1, \dots, x_n) \in X_1 \times \dots \times X_n \\ \phi(x_1, \dots, x_n) = y}} \left\{ T(\mu_{A_1}(x_1), \dots, \mu_{A_n}(x_n)) \right\} \quad (2.45)$$

Par le biais du principe d'extension, on peut par exemple obtenir une extension de l'addition (+) sur les ensembles flous, définie par :

$$\mu_{(A_1 \oplus A_2)}(y) = \sup \left\{ \min(\mu_{A_1}(x_1), \mu_{A_2}(x_2)) \mid (x_1, x_2) \in X_1 \times X_2 \wedge x_1 + x_2 = y \right\} \quad (2.46)$$

La figure 2.21 montre l'exemple d'une addition étendue des deux ensembles flous $\tilde{2}$ et $\tilde{3}$.

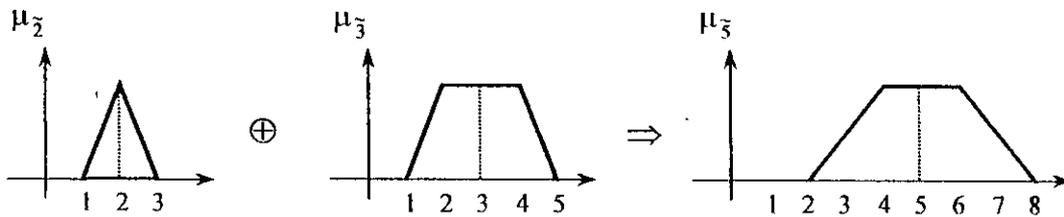


Figure 2.21 : Exemple d'addition étendue de deux ensembles flous

II.5.5. Les systèmes d'inférence flous

Un système flou est un système dans lequel on utilise la logique floue avec les outils mathématiques correspondants. La sortie peut être une conclusion, une décision ou une action de commande [Kel97].

Les ensembles flous sont introduits de plusieurs façons dans les systèmes :

1. *Dans la spécification des paramètres du système* : Un système est défini par une équation différentielle (algébrique ou en différence) dans laquelle les paramètres sont des nombres flous au lieu de nombres réels. Ceci est une manière de représenter les incertitudes paramétriques du système.

Exemple : $y = \tilde{3}x_1 + \tilde{5}x_2$; où $\tilde{3}$ et $\tilde{5}$ sont des nombres flous ("environ 3" et "environ 5").

2. *Dans la description du système* : Un système peut être entièrement défini par une collection de règles floues de type "If-Then" (ou Si-Alors) ou par une relation floue.

Exemples : *Si* la puissance est élevée, *Alors* la variation de température sera élevée.

Si la pression est très basse, *et Si* la température est assez élevée, *et Si* le débit est moyen, *Alors* ouvrir légèrement la vanne.

Si les cours montent assez rapidement, *et Si* la montée ne dure que depuis peu de temps, *et Si* les fluctuations sont assez faibles, *Alors* acheter beaucoup.

Un *système d'inférence flou* est un système basé sur les concepts de la théorie de la logique floue, des règles floues de type "If-Then", et du raisonnement flou. Il a été appliqué dans plusieurs domaines tels que le contrôle automatique, la classification des données, l'analyse des données ...etc. A cause de sa nature multi-disciplinaire, il est connu par différents noms, entre autres : un système d'inférence flou [Jan92], système flou à base de règles [Bab98b], système expert flou, modèle flou [Tak85], contrôleur (ou régulateur) flou [Mam75][Lee90] ou simplement un système flou.

II.5.5.1. Configuration de base d'un système d'inférence flou

Un système d'inférence flou est constitué de quatre parties principales (Figure 2.22) à savoir : une base de connaissances, une interface de fuzzification (ou interface de quantification floue), un moteur d'inférence et une interface de défuzzification.

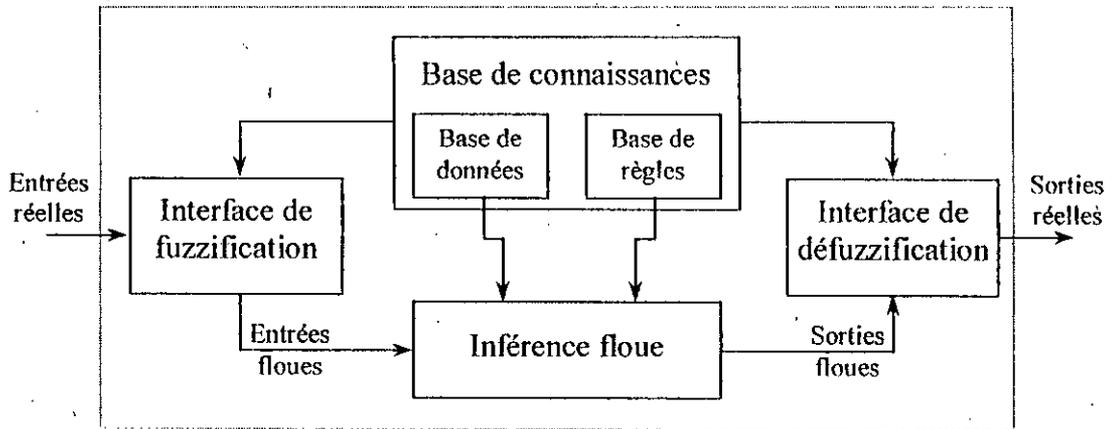


Figure 2.22 : Structure d'un système d'inférence flou

1. *L'interface de fuzzification* a pour but de déterminer les degrés d'appartenance des variables d'entrées aux ensembles flous correspondants. Elle reçoit des valeurs réelles (ou floues), représentant les entrées du système d'inférence flou, et les transforme en des termes linguistiques sous forme de valeurs entre 0 et 1.

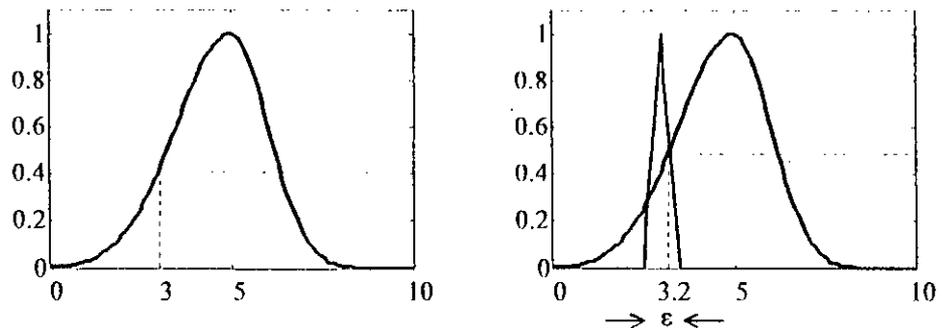


Figure 2.23 : Fuzzification, a) entrée réelle, b) entrée floue

La figure 2.23 montre la fuzzification d'une valeur exacte x_0 et d'une valeur x_0 entachée d'une incertitude ϵ .

2. *La base de connaissances* est constituée de deux blocs :

– *La base de données* : elle contient des informations qui caractérisent le système flou, telles que les univers de discours et les ensembles flous associés aux termes linguistiques d'entrées et de sorties.

– La base de règles : elle contient un nombre de règles de type If-Then de la forme :

$$\text{IF Condition (ou antécédent) THEN Conséquence} \quad (2.47)$$

La complexité des systèmes flous dépend fortement du nombre d'entrées et du nombre d'ensembles flous définis dans chaque univers de discours. La taille maximale de la base de règles croît exponentiellement avec le nombre des variables d'entrée du système flou :

$$\text{Nombre maximal des règles} = N \prod_{i=1}^n M_i \quad (2.48)$$

N , n et M_i sont respectivement le nombre de sorties, le nombre d'entrées et le nombre d'ensembles flous délimités par l'univers de discours de la $i^{\text{ème}}$ entrée.

La réduction de la complexité de la base des règles est un domaine de recherche très important pour les systèmes flous complexes [Kel97].

3. *L'unité d'inférence floue (unité de décision)* crée les variables floues de sortie en utilisant les opérations sur les fonctions d'appartenance.

Une fois l'antécédent évalué, son degré de validité (ou de certitude) est attribué à la conséquence de chaque règle. L'implication est fonction du degré de validité de l'antécédent β et de la fonction d'appartenance de la sortie $\mu_B(y)$. Plusieurs méthodes d'implication ont été proposées dans la littérature, qui conservent et généralisent l'implication de la logique booléenne [Bou95]. L'implication de Lukasiewicz définie par l'équation (2.49) en est un exemple.

$$\mu_B^*(y) = \text{imp}(\beta(x_0), \mu_B(y)) = \min(1 - \beta(x_0) + \mu_B(y), 1) \quad (2.49)$$

Malgré cela, la règle est en général modélisée par un opérateur de T-norme, bien que cette représentation ne généralise pas l'implication classique. Les exemples les plus classiques sont les implications de Larsen et de Mamdani définies par les opérations de produit ou de minimum respectivement (figure 2.24) :

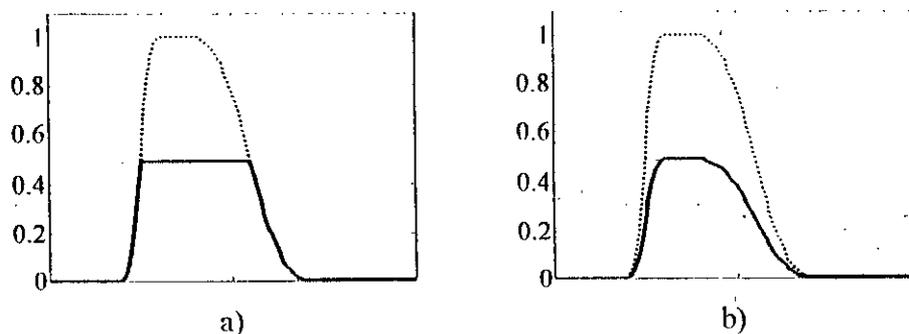


Figure 2.24 : Implication, a) par saturation, b) par produit

$$\begin{aligned}\mu_B'(y) &= \text{imp}(\beta(x_0), \mu_B(y)) = \beta(x_0) \cdot \mu_B(y) && \text{(produit)} \\ \mu_B'(y) &= \text{imp}(\beta(x_0), \mu_B(y)) = \min(\beta(x_0), \mu_B(y)) && \text{(saturation)}\end{aligned}\quad (2.50)$$

4. Agrégation : Dans un système flou, plusieurs règles sont actionnées en sortie, donc plusieurs ensembles flous de sortie sont concernés. L'agrégation détermine la manière de combiner les sorties des différentes règles pour construire un seul ensemble flou dans la conséquence.

Pour deux conséquences A et B , la combinaison de ces différentes règles se fait à l'aide de l'opérateur OU. Le résultat de l'agrégation est obtenu par plusieurs méthodes (généralement par une opération de T-conorme) [Mat99][Kel97], à savoir :

– *Méthode probabiliste* (figure 2.25a) :

$$C = \text{agg}(\mu_A(y), \mu_B(y)) = \mu_A(y) + \mu_B(y) - \mu_A(y) \cdot \mu_B(y) \quad (2.51)$$

– *Méthode de maximum* (figure 2.25b) :

$$C = \text{agg}(\mu_A(y), \mu_B(y)) = \max(\mu_A(y), \mu_B(y)) \quad (2.52)$$

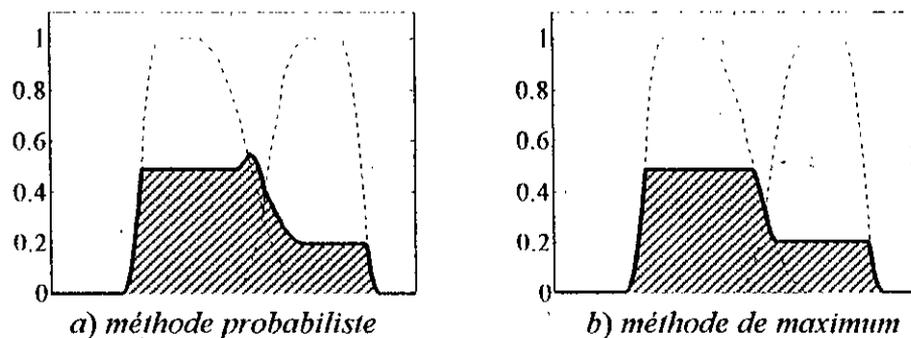


Figure 2.25 : Différentes méthodes d'agrégation

5. L'interface de défuzzification convertit l'ensemble flou résultant de l'agrégation en une seule valeur réelle, afin de donner plus de certitude à la sortie du système d'inférence flou.

Il existe plusieurs méthodes pour réaliser cette opération, parmi lesquelles :

– *La méthode du maximum* : Elle consiste à ne prendre en compte que les valeurs possédant un degré maximum d'appartenance à l'ensemble flou agrégé. Cette technique est simple à appliquer, mais peu employée, étant basée sur l'opérateur maximum qui privilégie une seule règle parmi celles qui sont présentes.

– *La méthode de la moyenne pondérée (méthode des hauteurs pondérées ou méthode de la moyenne des maximums)* : Celle-ci considère comme valeur de sortie la moyenne des valeurs préconisées par chaque règle, et pondérées par leur degré respectif de validité.

$$mp(y) = \frac{\sum_{i=1 \dots N} \bar{y}_i \cdot \mu_{B_i}(\bar{y}_i)}{\sum_{i=1 \dots N} \mu_{B_i}(\bar{y}_i)} \quad (2.53)$$

avec \bar{y}_i la valeur de sortie préconisée par la $i^{\text{ème}}$ règle, et qui peut correspondre au maximum de la fonction d'appartenance de l'ensemble flou B_i tel que :

$$\bar{y}_i = \sup_{y \in Y} (\mu_{B_i}(y)) \quad (2.54)$$

B_i : l'ensemble flou résultant de la $i^{\text{ème}}$ règle.

– *La méthode du centre de gravité* : Cette méthode, qui est la plus utilisée, génère le centre de gravité de la surface engendrée par la fonction d'appartenance de l'ensemble flou agrégé. L'abscisse du centre de gravité peut être déterminée par la relation suivante :

$$cdg(y) = \frac{\int_Y y \cdot \mu_{B_i}(y) dy}{\int_Y \mu_{B_i}(y) dy} \quad (2.55)$$

Cette méthode est très coûteuse en temps de calcul. De ce fait, certains préfèrent utiliser des formes simples de fonctions d'appartenance, telle que la forme triangulaire, dont les surfaces sont déterminées en fonction de leurs paramètres. Alors que d'autres préfèrent la méthode de défuzzification de la moyenne pondérée, pour sa simplicité et pour la facilité de sauvegarde des fonctions d'appartenance de sortie [Bou99b].

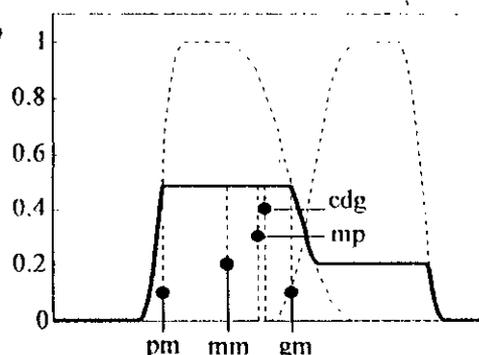


Figure 2.26 : Différentes méthodes de défuzzification

(*cdg* : centre de gravité, *mp* : moyenne pondérée, *pm* : petit maximum, *mm* : milieu des maxima, *gm* : grand maximum)

II.5.6. Les contrôleurs flous

Un contrôleur flou est un système d'inférence flou utilisé pour le réglage d'un système dynamique. Il a été introduit pour la première fois par E.Mamdani et Assilian en 1975 [Mam75] pour une régulation d'une machine à vapeur. Et ce n'est qu'à la fin des années 80 qu'il a connu un essor remarquable, en particulier au Japon [Hir95]. Il a été appliqué dans la commande automatique d'une machine à laver par Patsushita Electric Industrial, en janvier 1990.

L'origine du développement de la théorie de commande par logique floue a été le contrôle de processus basé sur la connaissance d'un opérateur humain qualifié, plutôt que sur des équations mathématiques. Les connaissances de l'opérateur sont schématisées par un ensemble de règles linguistiques de décision, dont la forme dépend essentiellement du processus à commander.

Des expériences ont montré que dans de nombreux cas les contrôleurs flous donnent des résultats meilleurs que ceux obtenus avec les techniques conventionnelles de commande. En particulier, le contrôleur flou apparaît utile quand les processus à commander présentent des difficultés d'analyse avec les techniques conventionnelles, ou quand leurs modèles sont jugés complexes, non définis ou incertains.

Dans ce qui suit, nous présenterons la contribution des contrôleurs flous dans la commande des systèmes dynamiques.

II.5.6.1. Différents types de contrôleurs flous

Dans cette partie, nous présenterons trois types de contrôleurs flous :

a) Contrôleurs PI-flou et PD-flou

Beaucoup de systèmes ont été commandés par des régulateurs classiques (PI, PD, IP, PID). Les paramètres sont déterminés en se basant sur la linéarisation de leurs modèles autour de leurs points de fonctionnement. Malheureusement, ces régulateurs sont dimensionnés à partir du modèle dynamique du système à contrôler. Par conséquent, une mauvaise modélisation, ainsi que la non-linéarité du système présentent l'inconvénient majeur de ce type de contrôleurs classiques, ce qui a permis à d'autres contrôleurs non-linéaires de voir le jour, entre autres les contrôleurs flous.

En considérant, comme variables d'entrées, l'erreur et la dérivée de l'erreur entre la consigne et la grandeur à réguler, les contrôleurs flous sont généralement vus comme étant des régulateurs PI- ou PD-flous [Dri93].

La figure 2.27 montre les structures de base des régulateurs PI- et PD-flous [Dri93] [Car94], où x_N est la valeur normalisée de la variable x .

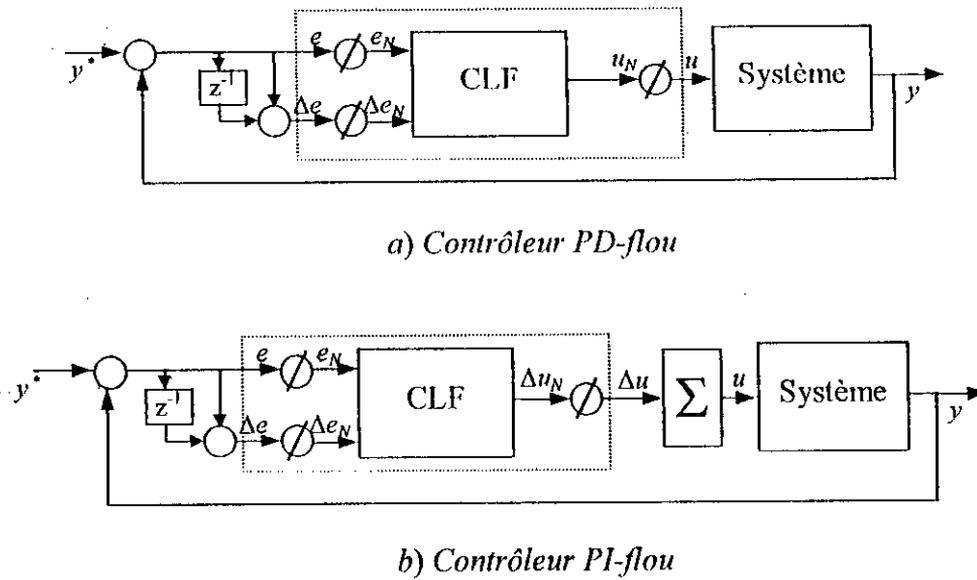


Figure 2.27 : Structure des régulateurs PI-flous et PD-flous

b) Contrôleurs adaptatifs flous

En général, une bonne adaptation des paramètres des régulateurs classiques, tels que les PID, améliore les caractéristiques dynamiques de la réponse du système commandé. Pour ce faire, un bloc supplémentaire jouant le rôle du superviseur flou est adjoint, permettant ainsi une pondération automatique de ces paramètres suivant les conditions de fonctionnement du système à contrôler (figure 2.28) [Car94][Kun94][Gha97][Yoo96].

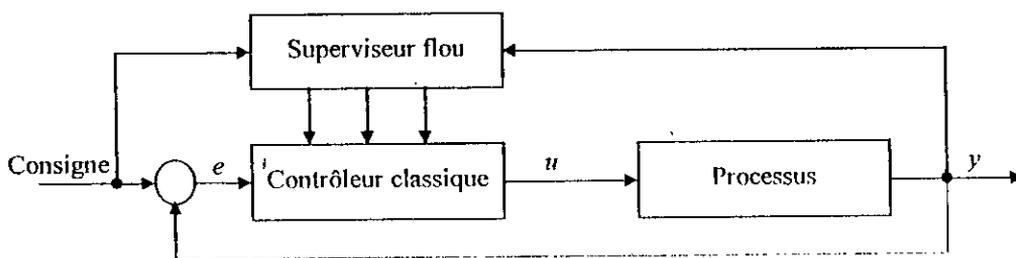


Figure 2.28 : Commande supervisée par logique floue

Une autre approche de la commande adaptative floue consiste en une régulation par adaptation des paramètres du contrôleur flou comme il est montré à la figure 2.29. Le mécanisme d'adaptation agit sur la phase de défuzzification, sur les ensembles flous ou sur la base de règles, selon le choix du concepteur [Dri93][Kim95][Tle99].

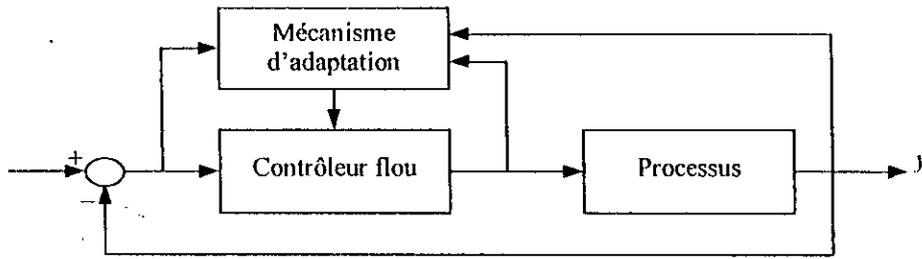


Figure 2.29 : Adaptation d'un contrôleur flou

c) Commande à structure variable floue

Nous avons vu, dans la commande par mode de glissement, que la réduction du phénomène de chattering dépend étroitement du choix de la fonction gain $K(x)$ de la partie discontinue donnée par l'équation (2.26). Pour cela, cette fonction gain est généralement choisie comme étant une relation linéaire en fonction de la distance entre la position de l'état et la surface de glissement définie dans l'équation (2.30). Il existe d'autres fonctions du gain $K(x)$, ayant des formes plus complexes [Bar00], et susceptibles de donner de meilleures performances, mais au prix d'un algorithme plus gourmand vis-à-vis du temps de calcul, d'où l'idée d'introduire le concept de la logique floue.

Une approche de commande à structures variables floue consiste à concevoir un système d'inférence flou ayant pour sortie le gain de la commande discontinue et pour entrées l'erreur et ses dérivées successives qui constituent la surface de glissement [Dri96]. La sortie de commande peut être décrite par l'équation suivante :

$$u = u_{eq} - K_{flou}(e, \dot{e}, \dots, e^{(n-1)}) \cdot \text{sign}(s(e, \dot{e}, \dots, e^{(n-1)})) \quad (2.56)$$

Pour un système du deuxième ordre, le gain K est généré par un système flou comme le montre la figure 2.30 [Li97]. Dans ce cas, la base de règles est construite en tenant compte de la condition d'atteignabilité.

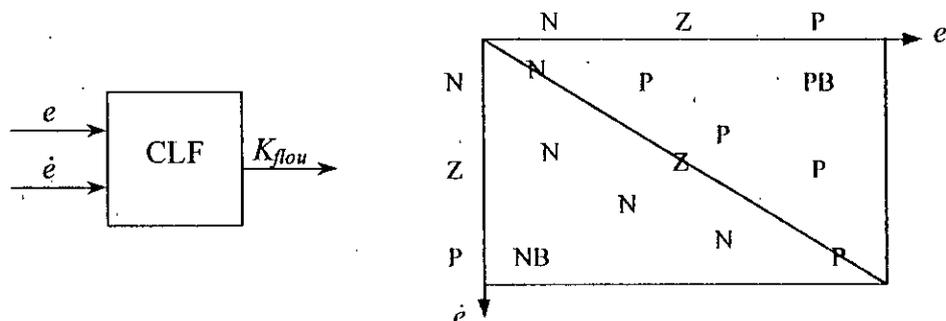


Figure 2.30 : Génération de la commande en tenant compte des erreurs successives des variables de sortie

Une approche de détermination du paramètre K_{flou} est donnée par la figure 2.31 [Li97] [Dri96]. Dans cette approche de commande, le système flou, possédant comme entrée unique la fonction de commutation s , est défini par la base de règles suivante :

$$\text{If } s \text{ est } F_s \text{ Alors } K \text{ est } F_K \quad (2.57)$$

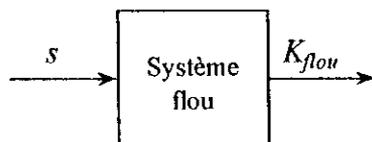


Figure 2.31 : Génération de la commande en tenant compte du calcul de la fonction de commutation

La commande à structures variables floue peut être basée sur le calcul de la fonction de commutation s et de sa dérivée \dot{s} . Le gain de la commande discontinue dépendra de ces deux grandeurs (figure 2.32) [Li97][Dri96][Hsu98].

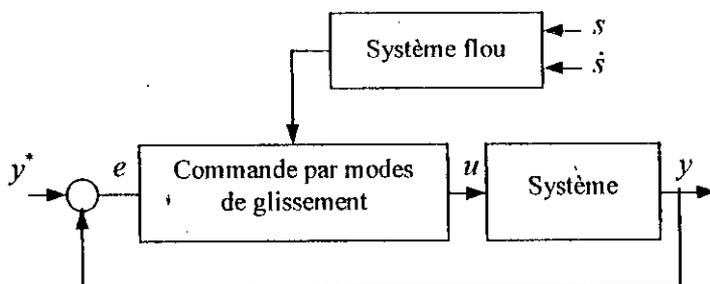


Figure 2.32 : Schéma de commande à structure variable floue en tenant compte du calcul de la fonction de commutation et de sa dérivée

II.5.7. Conception d'un contrôleur flou

Les contrôleurs flous peuvent être aussi classifiés selon la manière de les concevoir. La figure 2.33 illustre bien les principaux axes mis à la disposition du concepteur avant de fixer son choix sur l'utilisation de la logique floue dans la stratégie de commande.

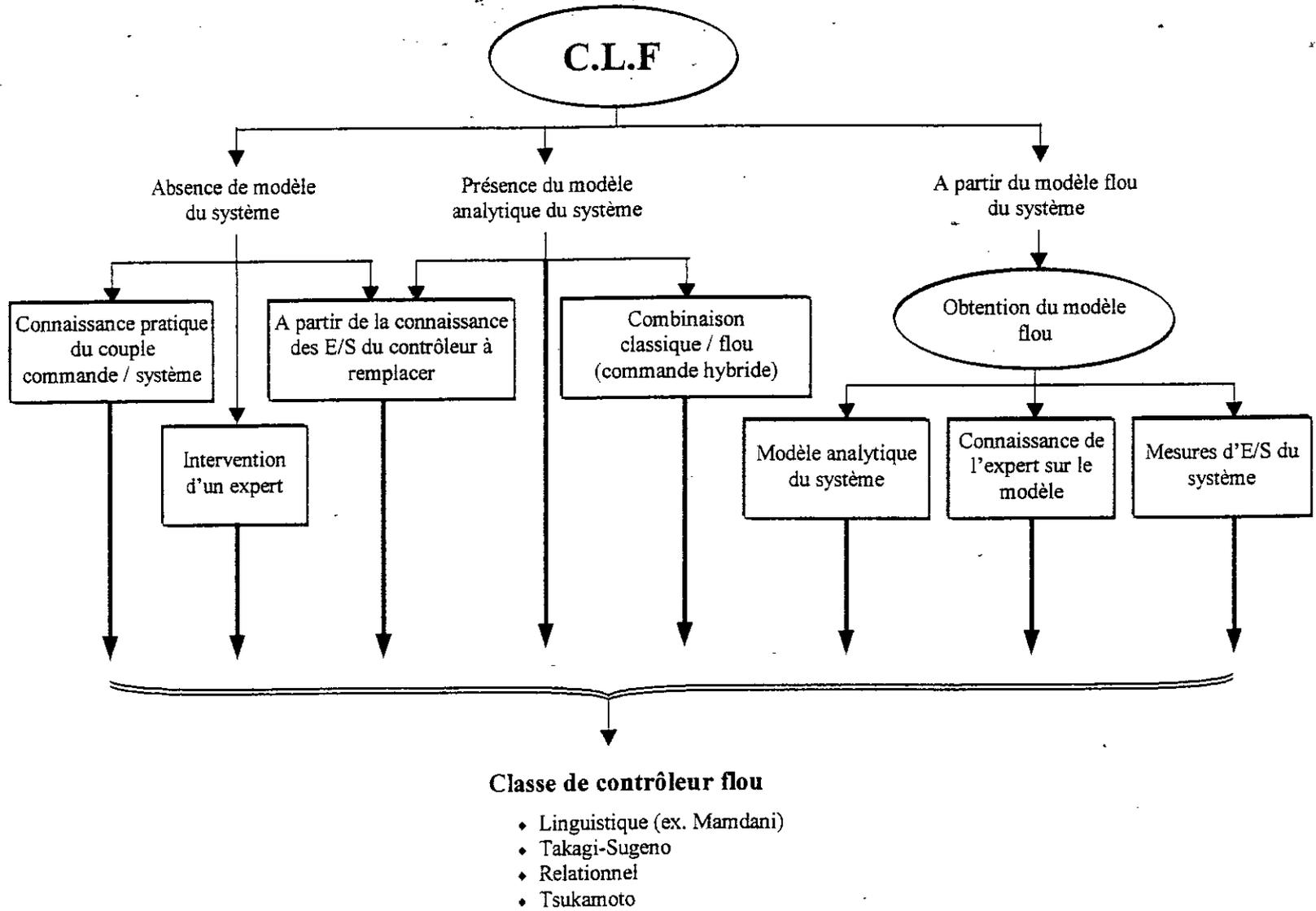


Figure 2.33 : Conception d'un contrôleur flou

II.5.7.1. Cas de l'absence du modèle du système

Au préalable, il faut étudier le système à régler et en établir une description adéquate. Dans ce cas, il ne s'agit pas d'une analyse proprement dite pour concevoir un modèle mathématique, mais plutôt de voir le comportement du système à régler vis-à-vis de la variation de la grandeur de commande.

Trois méthodes peuvent être utilisées, à savoir :

- Connaissance pratique (par identification) du couple : variable de commande / sortie correspondante du système.
- Prévion du comportement dynamique du système par l'intermédiaire d'un expert.
- À partir de la connaissance des entrées/sorties du contrôleur classique déjà existant.

II.5.7.2. Cas de la présence du modèle analytique du système

Il se subdivise en trois catégories :

- *Réglage du contrôleur flou par simulation*

Le contrôleur flou peut être conçu par simulation numérique en faisant appel au modèle mathématique du système original s'il existe. Ce qui facilite la mise en œuvre d'un contrôleur adéquat pour la stratégie de commande.

- *A partir de la connaissance des entrées/sorties du contrôleur classique à remplacer*

La connaissance du patron entrées/sorties du régulateur classique déjà existant peut faciliter et peut être utilisée pour la mise en œuvre du contrôleur flou.

Le but de cette méthode est d'avoir un système approximatif ayant des performances dynamiques meilleures que celles obtenues dans le cas classique du point de vue non-linéarité, simplicité d'implémentation et facilité d'ajustage [Dri96].

- *Commande hybride floue / classique*

Certaines stratégies de commande sont constituées d'un contrôleur hybride, c'est-à-dire que la loi de commande est une combinaison de deux parties distinctes, une partie analytique et une partie à base de règles floues. L'objectif d'un tel mixage est d'améliorer les performances (robustesse, stabilité, ...etc) et de permettre l'implémentation pratique dans le cas où celle-ci se trouve être difficile, voire impossible, par exemple dans le cas de la commande à structures variables quand la fonction gain a une forme non-linéaire complexe [Dri96].

II.5.7.3. A partir du modèle flou du système

La variable de commande u du système réel est obtenue à partir du modèle flou représentatif du système, et cela de deux manières différentes comme le montre la figure 2.34. Dans le cas de la figure 2.34a, l'existence du modèle flou facilite la conception du contrôleur flou du point de vue de la base des règles et des ensembles flous des entrées / sorties [Lyg94][Gha97][Kan98][Bra96b]. Alors que dans la figure 2.34b, le modèle flou sert à l'adaptation des paramètres du contrôleur, qui peut être éventuellement réel [Fis98]. Cela, après une phase de défuzzification des grandeurs de sortie du modèle, contrairement au premier cas, où cette phase s'effectue après le contrôleur flou.

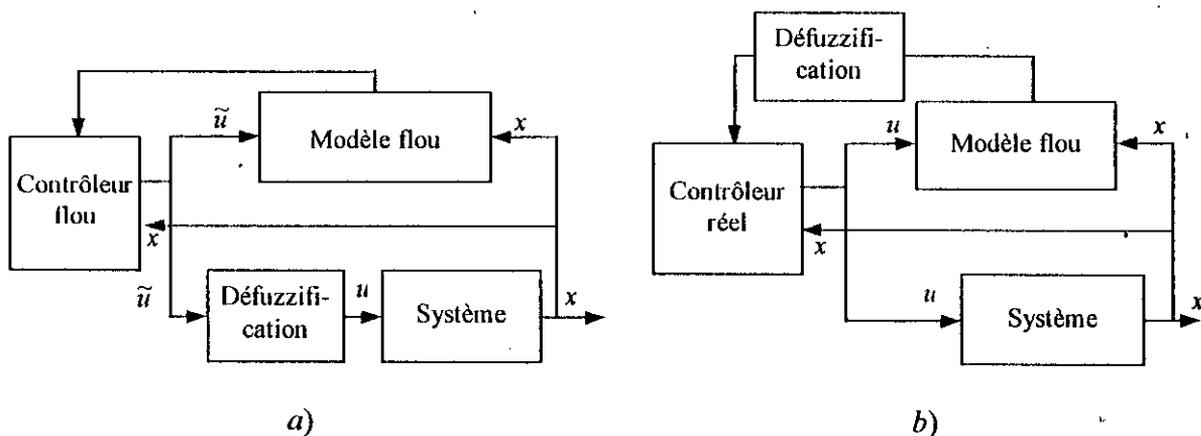


Figure 2.34 : Structure de base d'un système de commande basé sur un modèle flou

– *Élaboration du modèle flou à partir de la connaissance d'un expert*

L'expert, qui est une personne familière au système, peut élaborer le modèle flou sur la base de connaissances préalables du comportement du système en vue de sa commande.

– *Élaboration du modèle flou à partir des mesures d'entrées / sorties du système*

Pour une élaboration en off-line du modèle flou, on procède par des mesures soit sur des expériences pratiques sur le comportement du système à réguler, soit sur des simulations (si le modèle analytique existe). Néanmoins, avec cette méthode, un inconvénient majeur pourrait se présenter si, dans la phase du traitement, les données sont mal choisies. En effet, si ces dernières sont tronquées ou sont trop dispersées, on pourrait avoir une mauvaise extrapolation ou une mauvaise interpolation, selon le cas.

Par conséquent, l'identification en temps réel trouve ses avantages dans les commandes adaptatives floues indirectes [Tle99] qui ne demandent pas de connaissances structurelles ou paramétriques sur le modèle du système.

– *Élaboration du modèle flou à partir du modèle analytique du système*

Cette méthode consiste à concevoir le modèle flou du système à partir de la connaissance de l'expression analytique du modèle d'origine. Le modèle flou a pour but de remplacer ce dernier par un système d'équations floues représentatif [Gha96a]. Dans la technique de commande que nous avons adoptée pour notre travail, les incertitudes sont prises en compte par le modèle flou. Ce qui a l'avantage d'une meilleure robustesse dans la stratégie de commande floue.

II.6. Conclusion

Dans ce chapitre, nous avons présenté plusieurs stratégies classiques de commande du moteur asynchrone, à savoir la commande à flux orienté, la commande par placement de pôles basée sur un retour d'état linéarisant le modèle, et la commande à structure variable. L'application de cette dernière commande a permis de mettre en évidence une simplicité de conception et une supériorité des performances, relativement à celles obtenues avec les deux commandes précédentes.

En plus, nous avons introduit la notion de base de la logique floue, les systèmes d'inférence flous et les contrôleurs flous qui ont vu le jour ces dernières années, et qui ont prouvé leurs aptitudes à remplacer les méthodes classiques en donnant des performances comparables, voire nettement meilleures. Par la suite, nous avons procédé à une étude synthétique qui a pour but d'aider le concepteur de prendre le choix convenable lors de l'élaboration du contrôleur flou approprié à son système. Ce choix se fait selon l'absence ou la présence de l'expression analytique du modèle, des données expérimentaux, de son expérience, ou à partir du modèle flou du système que nous avons retenu.

Avant de s'engager dans l'étude de la conception des contrôleurs flous basés sur les modèles flous, nous allons présenter dans le prochain chapitre des notions générales sur les modèles flous et leur élaboration.

Chapitre III

Modélisation Floue des Systèmes Incertains

III.1. Introduction

L'approche traditionnelle de la modélisation est basée sur une parfaite connaissance du système à étudier, ainsi que sur des connaissances mathématiques appropriées au développement du modèle.

Pour des processus grossièrement connus, plusieurs méthodes ont été développées pour la mise en œuvre de leurs modèles et ceux de leurs contrôleurs. Parmi ces méthodes, on peut citer la modélisation floue qui englobe les techniques de modélisation basées sur la logique floue.

Les modèles flous sont des modèles qui utilisent des règles de type "If-Then" et des opérateurs logiques pour établir des relations qualitatives entre les diverses variables du modèle. La nature de la base de règles des modèles flous permet l'utilisation d'informations exprimées en langage naturel et permet une transparence dans l'analyse et l'interprétation de ces modèles. D'autre part, les modèles flous peuvent approximer une large classe de systèmes non-linéaires avec un certain degré d'exactitude.

Comparé aux autres techniques d'approximation non-linéaires, telles que les réseaux de neurones, le système flou fournit une représentation plus transparente au concepteur et peut donner une interprétation linguistique sous forme de règles [Bab98b].

Dans ce chapitre, nous allons présenter les notions de base et la classification des modèles flous. Notre investigation sera ensuite consacrée à une approche de modélisation floue des systèmes incertains. Le but de cette approche est d'interpréter un système dynamique incertain comme un système dynamique flou, et de fournir le modèle flou correspondant. Ce dernier sera décrit par une équation différentielle floue, servant pour la conception d'un contrôleur flou robuste.

III.2. *Évaluation des systèmes réels et flous*

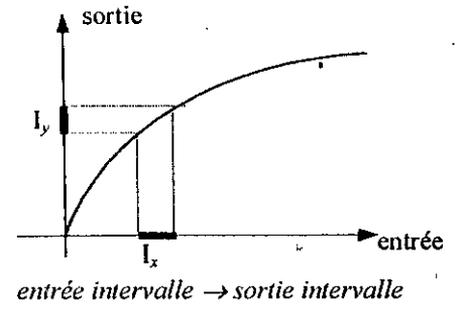
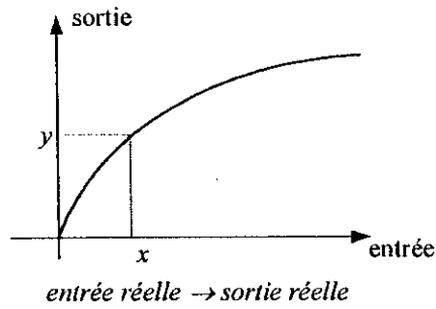
On distingue trois différents types de systèmes [Bab98b], comme le montre la figure 3.1 :

- ☛ les systèmes décrits par des fonctions réelles, qui sont généralement des systèmes certains
- ☛ les systèmes décrits par des fonctions intervalles
- ☛ les systèmes décrits par des fonctions floues.

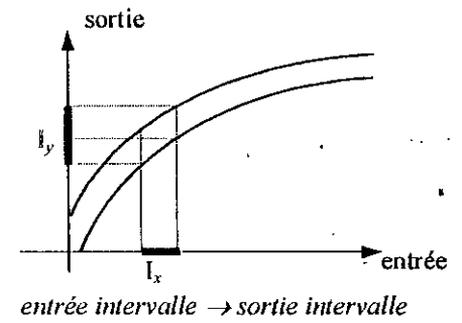
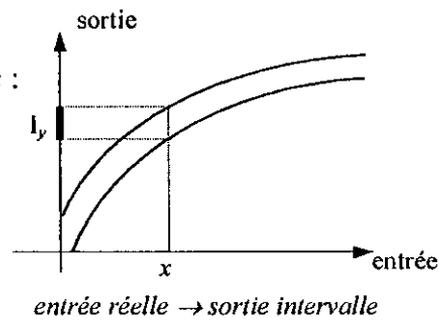
La différence entre la fonction intervalle et la fonction floue réside dans le fait que dans le premier cas, l'appartenance d'un point donné de l'espace est absolu, alors que pour le second, chaque point appartient à la fonction avec un degré de validité spécifique pour chaque point.

D'une manière générale, les systèmes flous peuvent être vus comme une généralisation de systèmes dont les variables d'entrée, les variables d'état et les paramètres sont des intervalles de valeurs réelles.

Fonction réelle :



Fonction intervalle :



Fonction floue :

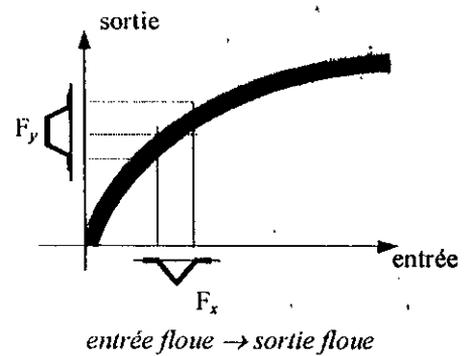
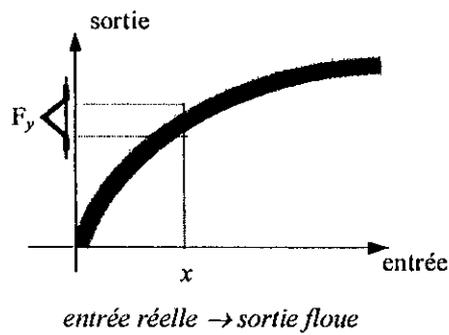


Figure 3.1 : Évaluation d'une fonction réelle, intervalle ou floue pour des arguments réels, intervalles ou flous

III.3. Les modèles flous à base de règles

Dans les modèles flous à bases de règles, les relations entre les variables d'entrée et de sortie sont assurées par des règles floues de type "If-Then" définies sous la forme suivante :

$$\text{IF } \tilde{x}_i \text{ is } F_x \text{ THEN } \textit{conséquence} \quad (3.1)$$

\tilde{x} : une variable linguistique (antécédent). Elle peut être considérée comme étant une variable réelle ou une variable floue (selon la prise en considération, ou non, des incertitudes sur la mesure ou l'estimation de x).

F_x : un ensemble flou (ou un terme linguistique).

Pour un modèle flou multivariable :

\tilde{x} : est un vecteur de variables réelles ou un vecteur de variables floues ou une région floue variable.

F_x : est un ensemble flou obtenu par un produit cartésien d'ensembles flous ayant chacun une fonction d'appartenance ou bien, d'une manière plus générale, par une région floue définie par une *fonction d'appartenance multivariées* $\mu_x(x)$ donnée dans l'intervalle $[0,1]$ comme suit :

$$\mu_x(x) : x \in \mathbb{R}^n \rightarrow [0,1] \quad (3.2)$$

Dans des cas particuliers, cette fonction d'appartenance multivariées peut être déterminée par des combinaisons logiques (min, max, prod, ...) entre des fonctions d'appartenance monovariées. Des opérateurs flous, tels que la conjonction, la disjonction et la négation (complément), sont utilisés pour combiner les termes flous des entrées.

Par exemple, une région floue F_x ayant une fonction d'appartenance $\mu_{F_x}(\cdot)$ de deux variables x_1 et x_2 définie par :

$$\mu_{F_x}(x) = \exp\left(\frac{-(x_1^2 + x_2^2)}{\sigma^2}\right) \quad (3.3)$$

elle est déterminée par une conjonction de deux ensembles flous monovariées F_{x_1} et F_{x_2} tels que :

$$\mu_{F_{x_1}}(x_1) = \exp\left(\frac{-x_1^2}{\sigma^2}\right) \quad \text{et} \quad \mu_{F_{x_2}}(x_2) = \exp\left(\frac{-x_2^2}{\sigma^2}\right) \quad (3.4)$$

La fonction d'appartenance de F_x est constituée par une opération de T-norme (produit) entre $\mu_{F_{x_1}}(x_1)$ et $\mu_{F_{x_2}}(x_2)$. Les ensembles flous F_{x_1} et F_{x_2} sont des projections de la région floue F_x sur les axes des x_1 et x_2 respectivement (Figure 3.2).

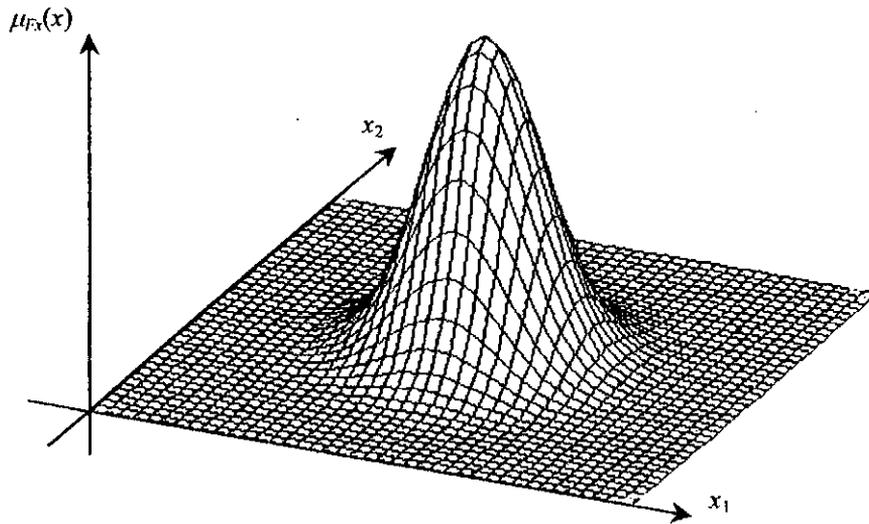


Figure 3.2 : Exemple de fonction d'appartenance gaussienne à deux dimensions

III.3.1. Partitionnement flou de l'espace d'entrée

Concernant l'antécédent du modèle flou, trois types de partitionnement peuvent être considérés [Bab98b][Jan92][Jan95] :

III.3.1.1. Partitionnement homogène

Soit un système flou, constitué de K règles, sous la forme conjonctive :

$$\mathfrak{R}_i: \text{IF } x_1 \text{ is } F_{x_1}^i \text{ AND } \dots \text{ AND } x_n \text{ is } F_{x_n}^i \text{ THEN } \dots \quad (3.5)$$

On remarque que ce modèle est un cas particulier de la forme générale (3.1), avec l'ensemble flou F_x le produit cartésien des ensembles flous $F_{x_j}^i$; ($j : 1, \dots, n$).

Dans ce cas, le degré de validité de l'antécédent de la règle, pour un vecteur d'entrée réel, $x = (x_1, \dots, x_n)^t$ sera calculé par :

$$\beta_i = \mu_{F_x^i}(x_1, x_2, \dots, x_n) = \mu_{F_{x_1}^i}(x_1) \wedge \mu_{F_{x_2}^i}(x_2) \wedge \dots \wedge \mu_{F_{x_n}^i}(x_n) \quad (3.6)$$

$\beta_i (i:1, \dots, K)$: le degré de validité de l'antécédent de la règle \mathfrak{R}_i .

\wedge : opérateur de la norme T-norme (minimum, produit, ...).

L'ensemble des règles, dans ce cas, divise l'espace du vecteur d'entrées en des régions floues dont chacune est considérée comme étant le produit cartésien des ensembles flous des entrées correspondantes (Figure 3.3).

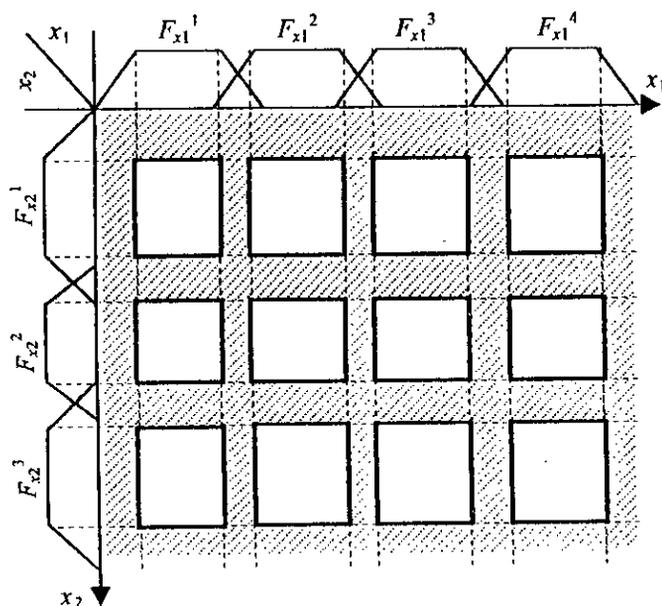


Figure 3.3 : Partition floue homogène de l'espace d'entrée schématisée par les noyaux des ensembles flous

Le nombre maximal des règles destinées à couvrir l'espace global du vecteur d'entrée est donné par :

$$K = N \prod_{i=1}^n M_i \quad (3.7)$$

avec

N : nombre de variables de sortie

M_i : nombre d'ensembles flous pour l'entrée x_i

n : nombre de variables d'entrée (la dimension de l'espace d'entrée)

III.3.1.2. Partitionnement par concaténation d'ensembles flous

Afin de réduire la complexité de la base de règles, on peut concaténer certaines régions floues de la partition homogène par des conjonctions, des disjonctions et/ou des négations sur les ensembles flous de l'antécédent.

Par exemple, en se référant à la figure 3.4, la région floue $F_x^{(1)}$ est obtenue en concaténant les deux régions floues $F_{x_1}^1 \times F_{x_2}^1$ et $F_{x_1}^1 \times F_{x_2}^2$.

La règle sera alors décrite par :

$$\text{IF } x_1 \text{ is } F_{x_1}^1 \text{ AND } x_2 \text{ is not } F_{x_2}^3 \text{ THEN ...} \quad (3.8)$$

La fonction d'appartenance de cette région floue est obtenue par l'expression suivante :

$$\mu_{F_x^{(1)}}(x) = \mu_{F_{x_1}^1}(x_1) \wedge (1 - \mu_{F_{x_2}^3}(x_2)) \quad (3.9)$$

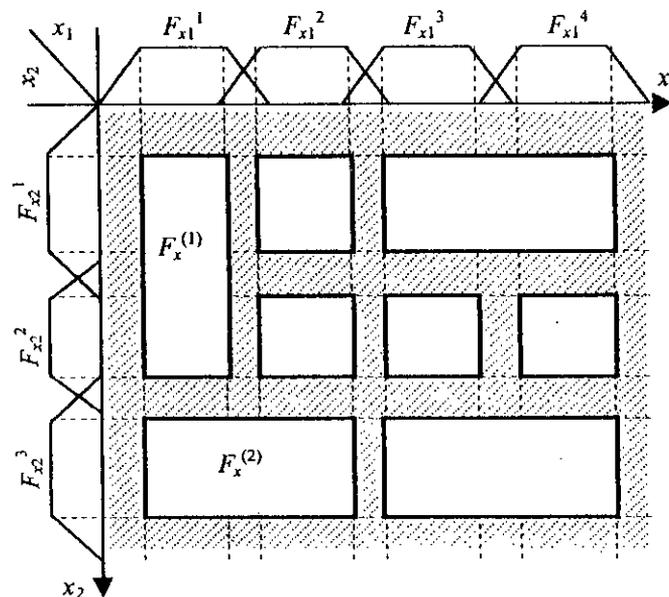


Figure 3.4: Partition floue obtenue par concaténation d'ensembles flous d'une partition homogène

L'avantage de cette nouvelle structure est qu'elle permet de réduire la cardinalité de la partition floue de l'espace d'entrées, donc la réduction de la complexité du modèle.

III.3.1.3. Partitionnement optimal

Dans le cas le plus général, il n'y a pas de restriction sur la forme des régions floues (Figure 3.5), lesquelles sont définies par des fonctions d'appartenances multivariables.

Le nombre de régions floues nécessaire pour couvrir l'espace du vecteur d'entrées peut être inférieur à celui des cas précédents. Pour les systèmes complexes multivariables, cette partition est la représentation la plus efficace.

Un point donné dans cet espace, pour une région floue donnée, fournit en même temps une information sur le degré d'appartenance à cette région de l'entrée x_1 et sur celui de l'entrée x_2 .

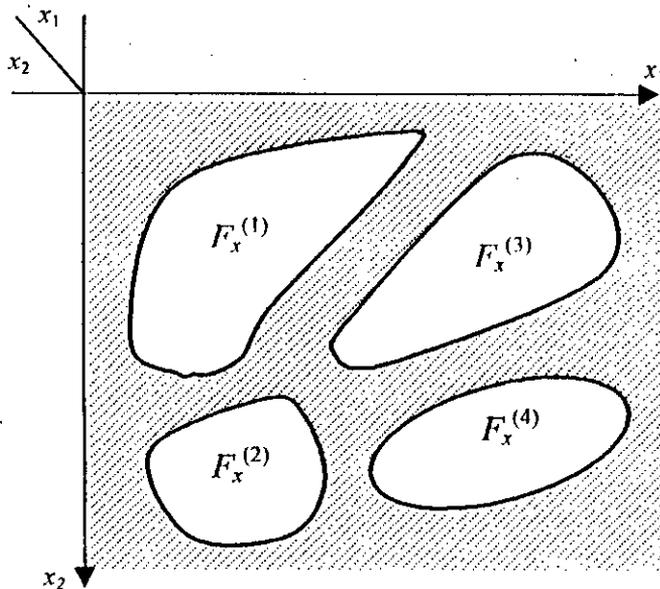


Figure 3.5 : Partition floue optimale

III.3.2. Différents types de modèles flous

Suivant la forme de la conséquence, on distingue quatre types de modèles flous :

- *Modèle flou linguistique* : dans lequel la conséquence est un ensemble ou une région floue F_y .
- *Modèle flou relationnel* : c'est une généralisation du modèle flou linguistique. Dans ce cas, il y a une relation floue entre les termes de la conséquence et de l'antécédent.
- *Modèle flou de type Takagi-Sugeno* : la conséquence de chaque règle est une fonction réelle.
- *Modèle flou de Tsukamoto* : la conséquence est un ensemble flou dont la fonction d'appartenance est monotone.

III.3.2.1. Le modèle flou linguistique

La forme générale d'un modèle flou linguistique basé sur des règles de type "If-Then" est donnée par :

$$\mathfrak{R}_i : \text{IF } \tilde{x} \text{ is } F_x^i \text{ THEN } \tilde{y} \text{ is } F_y^i \quad (3.10)$$

avec :

\mathfrak{R}_i : la $i^{\text{ème}}$ règle ; $i = 1, \dots, K$

\tilde{x} : variable linguistique d'entrée (antécédent) correspondante au vecteur des variables d'état et/ou des variables d'entrée, $x \in \mathbb{R}^n$.

\tilde{y} : variable linguistique de sortie (conséquence) correspondante au vecteur de sortie $y \in \mathbb{R}^q$.

F_x^i et F_y^i : sont des ensembles ou des régions floues constantes.

Les fonctions d'appartenance correspondantes à l'antécédent et à la conséquence sont définies par des fonctions multivariées :

$$\begin{aligned} \mu_{F_x^i}(x) : x \in \mathbb{R}^n &\rightarrow [0,1] \\ \mu_{F_y^i}(y) : y \in \mathbb{R}^q &\rightarrow [0,1] \end{aligned} \quad (3.11)$$

Mécanisme d'inférence

Le mécanisme d'inférence d'un modèle flou linguistique peut être décrit par les étapes suivantes :

– *Calcul du degré de validité de chaque règle :*

Supposons que le vecteur x est une variable linguistique connue par un ensemble flou A . Le degré d'appartenance de ce vecteur à F_x^i ($i = 1, \dots, n$) correspond au degré de validité de l'antécédent (\tilde{x} is F_x^i) de la $i^{\text{ème}}$ règle et il est calculé comme suit (Figure 3.6) :

$$\beta_i = \max_{U_x} \left(\mu_A(x) \wedge \mu_{F_x^i}(x) \right) ; \quad i = 1, \dots, K \quad (3.12)$$

où U_x est l'univers de discours ou l'espace du vecteur x .

$\mu_A(x)$ est la fonction d'appartenance du vecteur x à l'ensemble flou A .

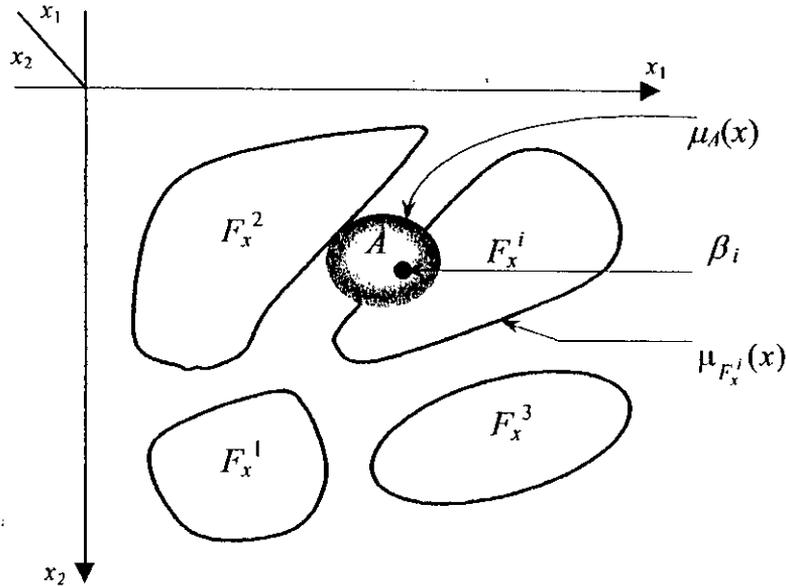


Figure 3.6 : Fuzzification ou obtention du degré de validité de l'antécédent d'une règle

Dans le cas où F_x^i et A sont des produits cartésiens des ensembles flous $F_{x_j}^i$ et A_j ($j = 1, \dots, n$) respectivement, c.-à-d. que l'univers de discours constitue une partition floue homogène (Figure 3.3), l'équation (3.12) peut être développée comme suit :

$$\mu_A(x) \wedge \mu_{F_x^i}(x) = \left(\mu_{A^1}(x) \wedge \dots \wedge \mu_{A^n}(x) \right) \wedge \left(\mu_{F_{x_1}^i}(x) \wedge \dots \wedge \mu_{F_{x_n}^i}(x) \right) \quad (3.13)$$

Suivant la propriété de l'associativité des opérateurs T-norme, l'équation (3.13) donne :

$$\begin{aligned} \mu_A(x) \wedge \mu_{F_x^i}(x) &= \left(\mu_{A^1}(x) \wedge \mu_{F_{x_1}^i}(x) \right) \wedge \dots \wedge \left(\mu_{A^n}(x) \wedge \mu_{F_{x_n}^i}(x) \right) \\ &= \bigwedge_{j=1:n} \left(\mu_{A^j}(x) \wedge \mu_{F_{x_j}^i}(x) \right) \end{aligned} \quad (3.14)$$

avec $\bigwedge_{j=1:n}$: opération de T-norme multivariable.

D'où l'expression du degré de validité de l'antécédent de la $i^{\text{ème}}$ règle :

$$\beta_i = \max_X \bigwedge_{j=1:n} \left(\mu_{A^j}(x) \wedge \mu_{F_{x_j}^i}(x) \right) = \bigwedge_{j=1:n} \max_X \left(\mu_{A^j}(x) \wedge \mu_{F_{x_j}^i}(x) \right) \quad (3.15)$$

Ou encore :

$$\beta_i = \bigwedge_{j=1:n} \beta_i^j \quad (3.16)$$

avec $\beta_i^j = \max_x \left(\mu_{A^j}(x) \wedge \mu_{F_i^j}(x) \right)$ comme illustré par la figure 3.7.

Le degré de validité de l'antécédent de la règle peut donc être déduit à partir d'une opération de T-norme sur les degrés d'appartenance du vecteur x aux ensembles flous qui constituent la partition floue homogène de l'espace d'entrées.

Dans certains cas, les règles peuvent être pondérées par des poids permettant d'augmenter ou de réduire l'importance d'une règle par rapport aux autres [Mat99]. Le degré de validité de la règle sera donc une opération de T-norme (produit en général) entre le degré de validité de l'antécédent β_i et le facteur de pondération correspondant.

– *Déduction de l'ensemble flou de chaque règle :*

La région floue de la sortie y est déduite par une opération d'inférence, qui peut être définie par une opération de T-norme [Bou95] :

$$\mu_{B_i}(y) = \beta_i \wedge \mu_{F_i^j}(y) \quad (3.17)$$

avec : $i = 1, \dots, K$

$y \in Y$; avec Y l'espace du vecteur de sortie y .

B_i : l'ensemble flou conséquent de la règle \mathfrak{R}_i .

– *Agrégation des ensembles flous de sortie des règles :*

Pour déduire une seule région floue de sortie résultant de l'ensemble des règles, une opération de T-conorme (max, somme, ...) peut être utilisée :

$$\mu_B(y) = S_{1 \leq i \leq K} \left(\mu_{B_i}(y) \right) ; y \in Y \quad (3.18)$$

$S(\cdot)$ opération de T-conorme.

Défuzzification

La région floue résultante de l'agrégation doit être défuzzifiée pour fournir des valeurs réelles à la sortie du modèle. La défuzzification est réalisée par une des méthodes mentionnées dans la section (§II.5) du chapitre précédent.

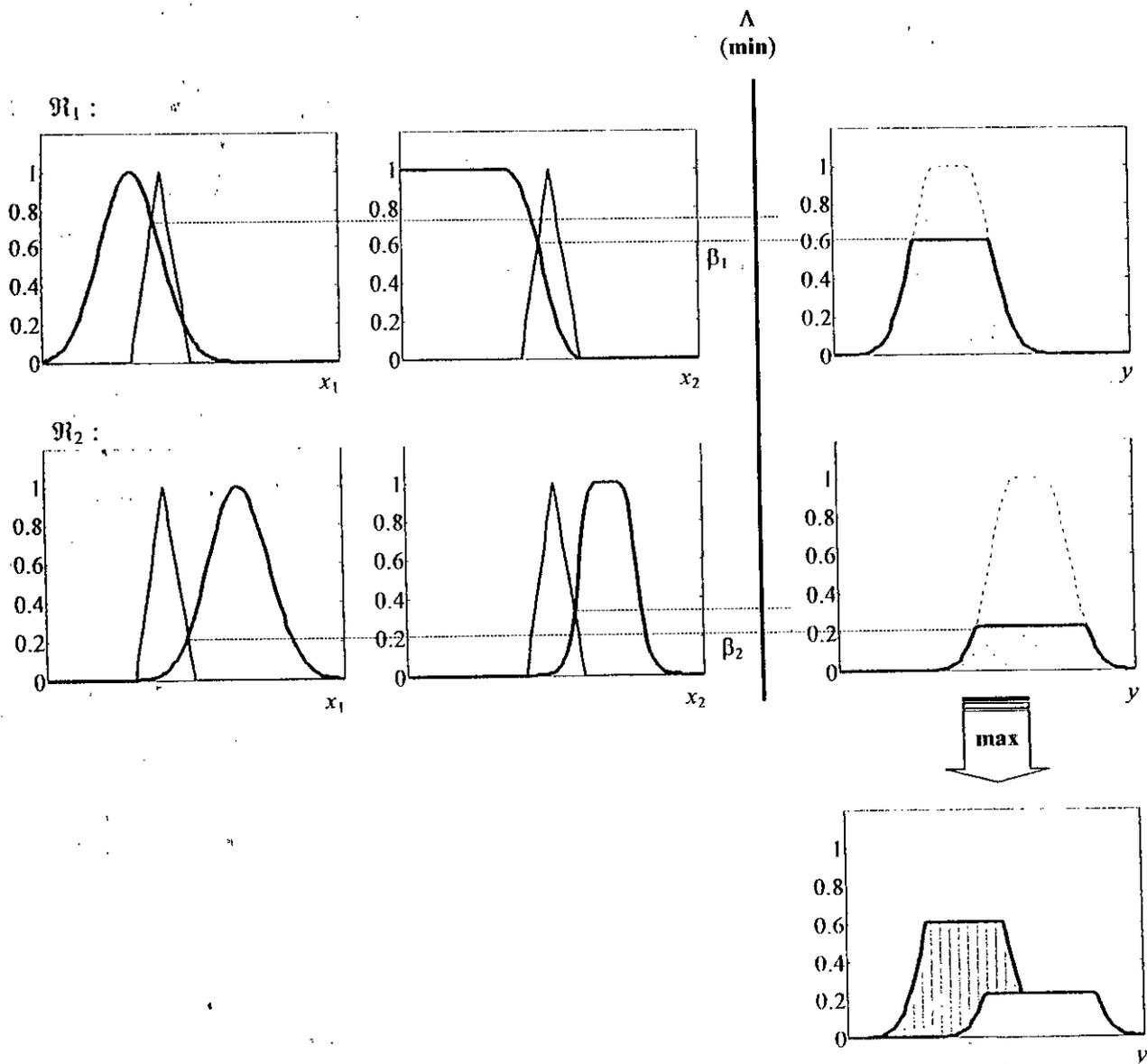


Figure 3.7 : Système d'inférence flou de Mamdani utilisant min et max pour les opérations ET et OU respectivement

III.3.2.2. Modèle flou relationnel

Un modèle flou relationnel est vu comme une généralisation du modèle flou linguistique, où les ensembles flous d'entrée et de sortie sont liés par des relations floues [Kre96][Bab98b].

Soit une base de règles d'un modèle flou linguistique définie sous la forme :

$$\mathfrak{R}^i : \text{IF } \tilde{x} \text{ is } F_x^i \text{ THEN } y \text{ is } F_y^i \quad (3.19)$$

$$x = [x_1, x_2, \dots, x_n]^t$$

Cette base de règle peut être représentée comme une relation \mathcal{R} entre le terme linguistique de l'antécédent F_x^i et celui de la conséquence F_y^i .

$$\mathcal{R} : F_x^i \times F_y^i \rightarrow \{0, 1\} \quad (3.20)$$

Chaque règle du modèle flou linguistique affecte un seul terme flou de sortie. C'est-à-dire que chaque relation $\mathcal{R}(F_x^i, F_y^i)$ est soit vraie, soit fausse (Figure 3.8).

Un modèle flou relationnel est obtenu par l'extension du modèle linguistique décrit par l'expression (3.20) :

$$\mathcal{R} : F_x^i \times F_y^i \rightarrow [0, 1] \quad (3.21)$$

Dans ce cas, chaque règle contient tous les termes de la sortie. Il est attribué à chaque terme un facteur de pondération (Figure 3.9).

Cette structure fournit plus de possibilités pour le réglage du modèle flou le plus adéquat à la description du système. Le modèle flou résultant présente l'avantage d'être facilement ajusté, sans avoir besoin de modifier ni sa structure, ni les formes ou le nombre des ensembles flous de la conséquence [Bab98b].

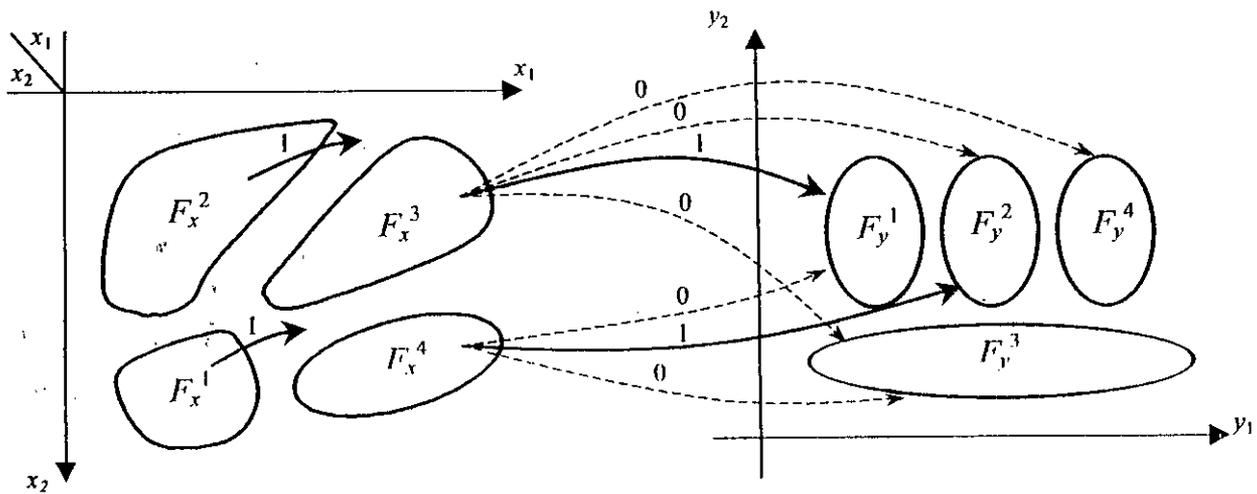


Figure 3.8 : Modèle flou linguistique (à deux entrées et deux sorties)

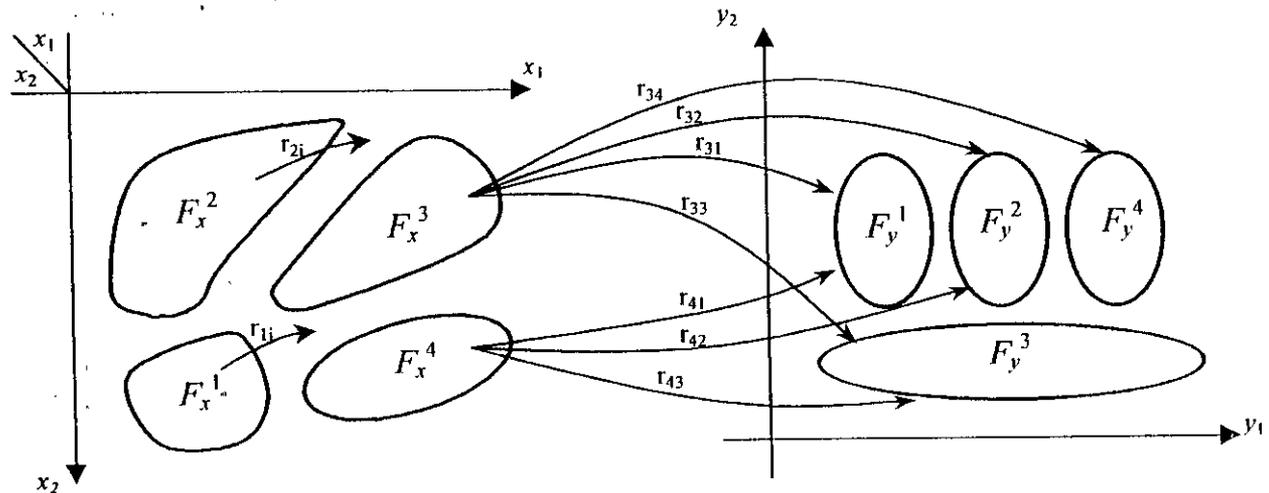


Figure 3.9 : Modèle flou relationnel (à deux entrées et deux sorties)

Mécanisme d'inférence

Le mécanisme d'inférence du modèle, pour un vecteur linguistique d'entrées, est donné par l'algorithme suivant :

1. Calcul du degré de validité de l'antécédent de chaque règle \mathfrak{R}_i :

$$\beta_i = \max_x (\mu_A(x) \wedge \mu_{F_x^i}(x)) ; i = 1, \dots, K \quad (3.22)$$

2. Application de la composition relationnelle $\omega = \beta \circ R$ donnée par :

$$\omega_j = S_{1 \leq i \leq K} (\beta_i \wedge r_{ij}) ; j = 1, \dots, M \quad (3.23)$$

\wedge : opération de T-norme

$S(\cdot)$: opération de T-conorme

Pour chaque sortie, les facteurs de pondération r_{ij} sont généralement constants et calculés en "off-line".

La région floue résultante de sortie B_j sera déduite par :

$$B_j = S(\omega_j, F_y^j) \quad (3.24)$$

3. Défuzzification de la région floue conséquente :

Vu les avantages du modèle relationnel, qui résident principalement dans la simplicité de l'ajustement de ses paramètres, une simple méthode de défuzzification ne demandant pas un grand temps de calcul est généralement considérée. Un cas particulier est proposé par Sugeno et Yasukawa en 1993 [Sug93][Kim97], et défini par :

$$y_{\text{moy}} = \frac{\sum_{j=1}^M \omega_j \cdot b_j}{\sum_{j=1}^M \omega_j} \quad (3.25)$$

avec $b_j = \int y \cdot \mu_{B_j}(y) \cdot dy / \int \mu_{B_j}(y) \cdot dy$ = centre de gravité de la surface engendrée par la fonction d'appartenance de B_j . Les centres de gravité, dans ce cas, sont calculés en "off-line".

III.3.2.3. Modèle flou de Takagi-Sugeno

Ce modèle décrit le comportement d'un système en utilisant des règles de type If-Then avec des propositions floues dans l'antécédent et des *fonctions réelles* dans la conséquence.

La fonction de sortie de chaque règle est valide localement dans la région floue correspondante à l'espace du vecteur d'entrées.

Une règle de Takagi-Sugeno (TS) a la forme générale suivante [Ped97] :

$$\mathcal{R}^i : \text{IF } x \text{ is } F_x^i \text{ THEN } y_i = f_i(x, a^i) ; i = 1, \dots, K \quad (3.26)$$

Le terme a^i constitue le vecteur des paramètres du $i^{\text{ème}}$ modèle local.

Dans la littérature, le vecteur d'entrée x n'est pas considéré comme étant une entrée linguistique, mais comme un vecteur de variables réelles.

Les fonctions f_i sont généralement de la même forme et ne diffèrent que par leurs paramètres. La forme la plus utilisée est la forme linéaire définie par [Tak85] :

$$\mathfrak{R}^i: \text{IF } x \text{ is } F_x^i \text{ THEN } y_i = a_i^t x + b_i; \quad i = 1, \dots, K \quad (3.27)$$

a_i : vecteur des paramètres.

b_i : valeur scalaire.

Ce modèle est appelé : *modèle linéaire de Takagi-Sugeno* (Figure 3.10).

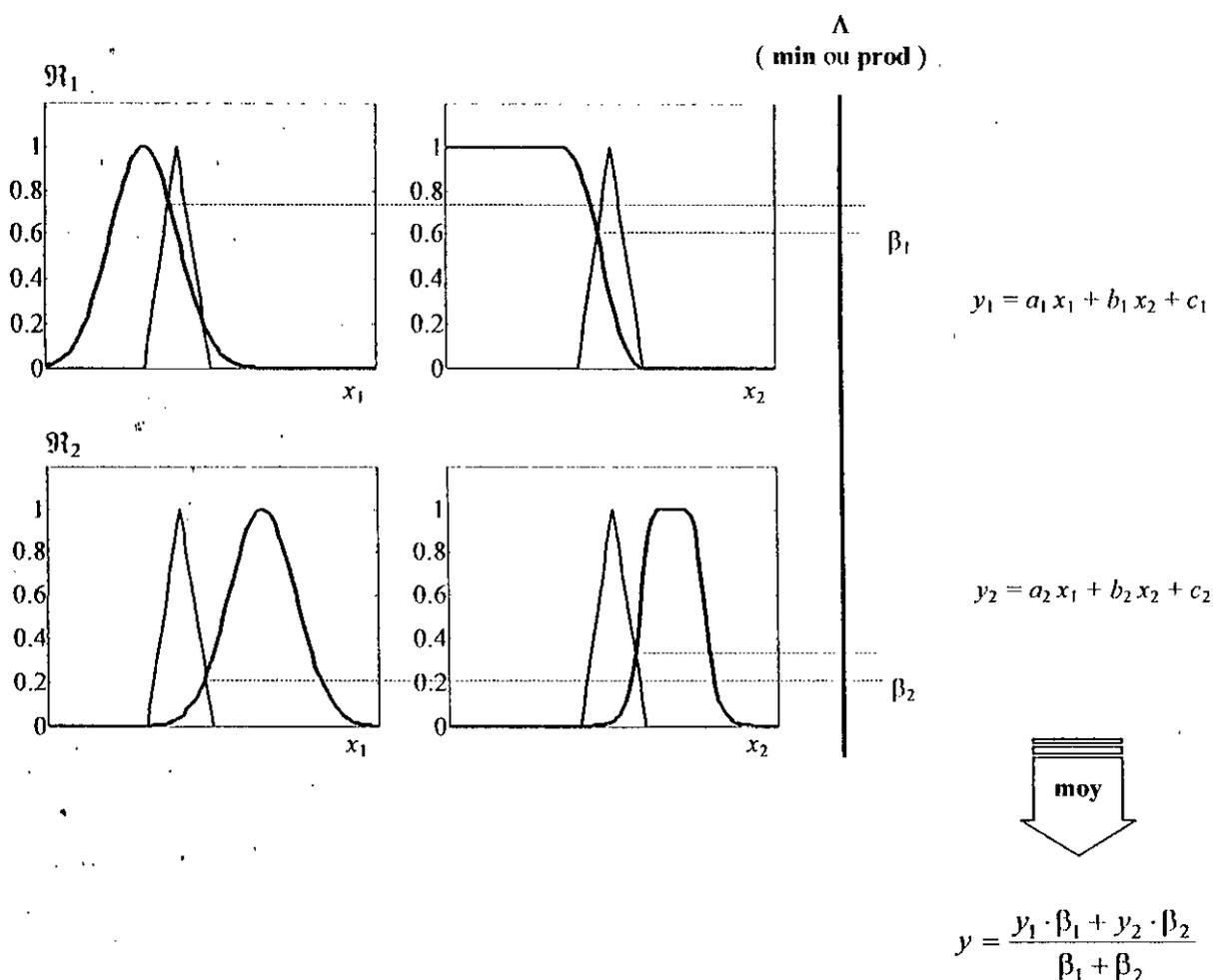


Figure 3.10 : Système d'inférence d'un modèle linéaire flou de Takagi-Sugeno

Mécanisme d'inférence

1. Calcul des degrés de validité de l'antécédent des règles " β_i " :

$$\beta_i(x = x_0) = \mu_{F_x^i}(x = x_0) \quad (3.28)$$

2. Calcul de la sortie :

$$y_{\text{moy}} = \frac{\sum_{j=1}^K \beta_j \cdot y_j}{\sum_{j=1}^K \beta_j} = \frac{\sum_{j=1}^K \beta_j \cdot (a_j^t x + b_j)}{\sum_{j=1}^K \beta_j} \quad (3.29)$$

Pour des degrés de validité normalisés, c.-à-d. :

$$\gamma_i = \frac{\beta_j}{\sum_{j=1}^K \beta_j} \quad ; \quad \sum_{j=1}^K \gamma_j = 1 \quad (3.30)$$

la sortie du modèle sera décrite par :

$$y_{\text{moy}} = \sum_{j=1}^K \gamma_j \cdot y_j = \sum_{j=1}^K \gamma_j \cdot (a_j^t x + b_j) \quad (3.31)$$

D'une autre manière, nous pouvons écrire :

$$y_{\text{moy}} = \sum_{j=1}^K \gamma_j \cdot a_j^t x + \sum_{j=1}^K \gamma_j \cdot b_j = \left(\sum_{j=1}^K \gamma_j \cdot a_j^t \right) x + \left(\sum_{j=1}^K \gamma_j \cdot b_j \right)$$

ou bien :

$$y_{\text{moy}} = a^t(x) \cdot x + b(x) \quad (3.32)$$

Comme les γ_i sont fonction du vecteur x , le modèle linéaire de TS peut être vu comme un modèle quasi-linéaire (un modèle linéaire avec des paramètres dépendant de x) avec :

$$a(x) = \sum_{j=1}^K \gamma_j(x) \cdot a_j \quad ; \quad b(x) = \sum_{j=1}^K \gamma_j(x) \cdot b_j$$

$a(x)$ et $b(x)$ sont des combinaisons linéaires des paramètres a_i et b_i . Cette propriété rend l'analyse des modèles de TS similaire à celle des systèmes linéaires [Bab98b][Sug88].

III.3.2.4. Modèle flou de Tsukamoto

Dans les modèles flous de Tsukamoto [Tsu79][Jan95], la conséquence de chaque règle, et pour chaque sortie, est représentée par un ensemble flou ayant une fonction d'appartenance monotone (Figure 3.11).

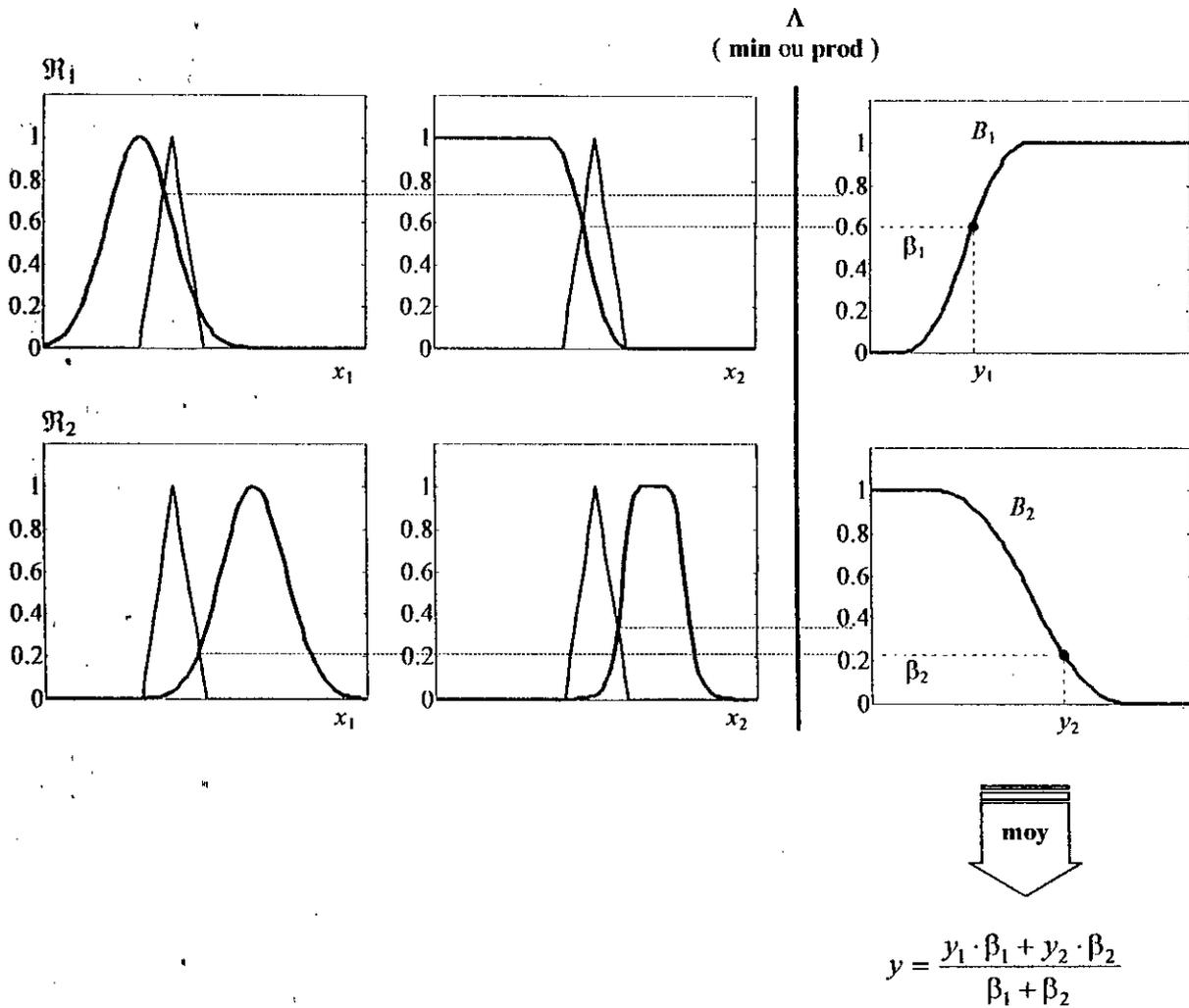


Figure 3.11 : Système d'inférence d'un modèle flou de Tsukamoto

III.4. *Élaboration d'un modèle flou*

Pour élaborer un modèle flou, deux étapes d'identification sont nécessaires : une identification structurelle et une identification paramétrique [Sug88][Bab98b]. Dans la première étape, on définit la base de règles et le type du modèle. Dans la seconde étape, on détermine les formes des ensembles flous et les paramètres des fonctions d'appartenance.

On peut citer trois sources d'information pour construire des modèles flous :

☛ *la connaissance linguistique préalable d'un expert humain :*

Le système dans ce cas peut être inconnu. L'expert doit avoir une connaissance sur le fonctionnement de son système et le comportement du contrôleur. De plus, lorsqu'il s'agit de problèmes complexes et multivariés, il est difficile d'établir directement une base de règles permettant le contrôle flou du système [Bor96].

☛ *les données de mesure :*

Pour plusieurs processus, les données sont disponibles comme des enregistrements ou des résultats d'identification. La conception et le réglage des modèles flous basés sur des données d'E/S est souvent appelé : *identification floue* [Sug88][Bra96a][Lu98][Man99].

☛ *la formulation mathématique grossière du modèle :*

Cette méthode utilise la connaissance disponible sur la modélisation physique du système et fournit une représentation floue du système tout en tenant compte des incertitudes. Dans ce cas, la base de règles est obtenue automatiquement, quel que soit le nombre de variables d'entrée du système flou [Gha96a][Ped97].

Dans ce qui suit, nous nous contentons de développer la troisième méthode, car elle est plus systématique et assure une meilleure imitation du système. En outre, elle s'adapte bien au problème qui nous concerne.

III.5. Modélisation floue linguistique des systèmes incertains

Considérons les systèmes incertains décrits par l'équation différentielle suivante :

$$\dot{x} = f(x, \eta) + g(x, \eta) \cdot u \quad (3.33)$$

où : $x = [x_1, x_2, \dots, x_n]^t \in \mathbb{R}^n$ le vecteur d'état du système,

$u(t) = [u_1, u_2, \dots, u_m]^t \in \mathbb{R}^m$ le vecteur d'entrée,

$\eta(t) \in \mathbb{R}^p$ le vecteur des incertitudes,

f, g sont respectivement un vecteur de dimension n et une matrice de fonctions de dimensions $n \times m$.

Toutes les incertitudes dans le modèle du système sont incluses dans $\eta(t)$. Ce dernier représente les incertitudes paramétriques, non paramétriques et structurales. La figure 3.12 montre la famille des systèmes dynamiques engendrée par les incertitudes η^i . A chaque valeur possible de $\eta(t) = \eta^i$ correspond un comportement du système.

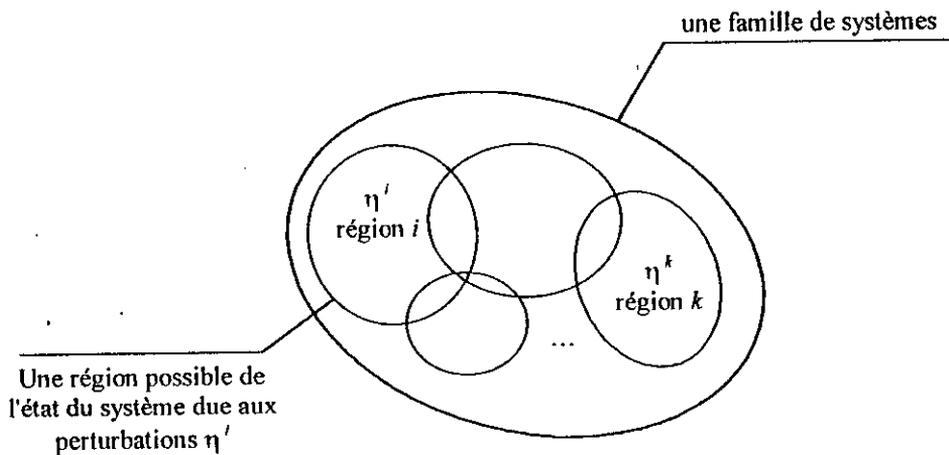


Figure 3.12 : Effet des incertitudes sur un système

Dans cette étude, les incertitudes non-paramétriques sont vues comme des dynamiques non modélisées et sont représentées comme des perturbations additives.

L'approche de modélisation floue de Ben-Ghaliya [Gha94][Gha96a][Gha97] consiste à produire une représentation basée sur des ensembles flous :

$$\tilde{\dot{x}} = \tilde{f}(\tilde{x}) \tilde{+} \tilde{g}(\tilde{x}) \tilde{\sim} \tilde{u} \quad (3.34)$$

où : \tilde{x} , \tilde{u} sont des vecteurs flous dont les composants sont des variables floues.

\tilde{f} , \tilde{g} sont respectivement un vecteur et une matrice contenant des fonctions floues.

$$\tilde{f}(x) = \begin{pmatrix} \tilde{f}_1(\tilde{x}_1, \dots, \tilde{x}_n) \\ \vdots \\ \tilde{f}_n(\tilde{x}_1, \dots, \tilde{x}_n) \end{pmatrix} ; \quad \tilde{g}(\tilde{x}) = [g_{ij}(\tilde{x}_1, \dots, \tilde{x}_n)]_{\substack{i=1, \dots, n \\ j=1, \dots, m}} \quad (3.35)$$

$\tilde{+}, \tilde{\cdot}$ sont respectivement l'addition et la multiplication étendue sur les variables floues [Dri93].

L'équation (3.34) est appelée *équation différentielle floue*. Les modèles décrits par cette équation sont appelés *modèles dynamiques flous*.

On note que l'élément η n'apparaît plus dans la nouvelle représentation du système ; son effet est pris en compte implicitement en utilisant la représentation par des ensembles flous.

III.5.1. Condition d'existence de modèle flou selon la méthode proposée

Concernant l'équation (3.33), nous considérons trois suppositions :

1. L'espace d'état $x = [x_1, \dots, x_n]^t$ appartient à un ensemble borné U_x , dont les bornes sont connues.

avec :

$$U_x = U_{x_1} \times U_{x_2} \times \dots \times U_{x_n} \subset \mathbb{R}^n. \quad (3.36)$$

2. L'élément des incertitudes $\eta(\cdot) : \mathbb{R}_+ \rightarrow \mathbb{R}^p$ est mesurable au sens de Lebesgue et ses valeurs $\eta(t)$ appartiennent à un ensemble borné $P \subset \mathbb{R}^p$, $\forall t \in \mathbb{R}_+$.
3. Le vecteur et la matrice de fonctions $f(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ et $g(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^n \times \mathbb{R}^m$ sont fortement Carathéodory, $\forall \eta \in P$ (Annexe D).

Les deux dernières suppositions assurent l'existence d'une possibilité de stabilité globale du système (3.33) soumis au contrôleur approprié. Ce dernier devient lui-même Carathéodory [Gha96b][Hal69].

III.5.2. Algorithme proposé de la modélisation floue

Pour définir les ensembles et les règles appropriés à l'équation différentielle floue (3.34), nous présentons l'algorithme constitué des étapes suivantes.

Étape 1 : Partitionnement de l'espace d'entrée

La première étape de la modélisation floue consiste à construire une partition floue homogène de l'univers de discours U_{x_i} de chaque composante x_i du vecteur d'état, $i = 1, \dots, n$. C'est-à-dire définir M_i ensembles flous $F_{x_i}^{k_i}$ pour chaque variable x_i . Cela est fait en partitionnant chaque univers de discours U_{x_i} en un nombre fini d'intervalles $I_{x_i}^{k_i}$ et en associant à chacun une fonction d'appartenance $\mu_{F_{x_i}^{k_i}}$.

$I_{x_i}^{k_i} \subset U_{x_i}$: représente le support du $k_i^{\text{ème}}$ ensemble flou de la variable linguistique x_i .

$k_i = K_{i1}, \dots, K_{i2}$; $M_i = K_{i2} - K_{i1} + 1$; avec k_i un nombre entier définit l'ensemble flou $F_{x_i}^{k_i}$.

L'univers de discours de la variable x_i doit vérifier l'équation suivante :

$$U_{x_i} \subseteq \bigcup_{k_i=K_{i1}}^{K_{i2}} I_{x_i}^{k_i} ; i = 1, \dots, n \quad (3.37)$$

Les ensembles flous $F_{x_i}^{k_i}$; $k_i = K_{i1}, \dots, K_{i2}$; forment une partition floue de U_{x_i} pour $i = 1, \dots, n$:

$$P_F(U_{x_i}) = \left\{ F_{x_i}^{k_i} \mid k_i = K_{i1}, \dots, K_{i2} \right\} \quad (3.38)$$

avec la cardinalité $P_F(U_{x_i}) = M_i$ pour $i = 1, \dots, n$.

La largeur de ce support et la cardinalité de la partition floue ainsi que la forme des fonctions d'appartenance sont choisies par le concepteur.

La variable x_i est alors associée à l'un des M_i ensembles flous $F_{x_i}^{k_i}$ dans l'univers de discours U_{x_i} . Le vecteur x peut correspondre à l'une des $\prod_{i=1}^n M_i$ combinaisons possibles du vecteur flou \tilde{x} , donc \tilde{x} représente un des $\prod_{i=1}^n M_i$ vecteurs d'état flous définis par :

$$F_x^{(k_1, \dots, k_n)} = \left[F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n} \right]^t \quad (3.39)$$

avec : $k_i = K_{i1}, \dots, K_{i2}$; $i = 1, \dots, n$.

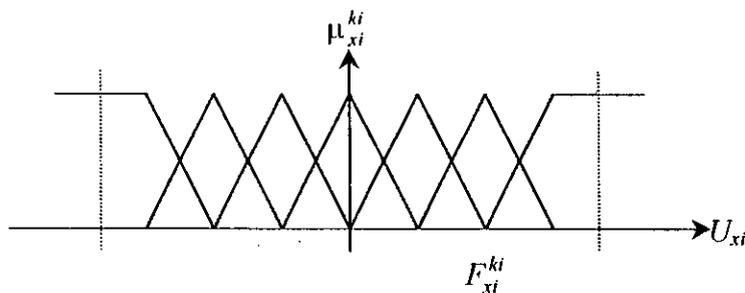


Figure 3.13 : Répartition des ensembles flous pour une variable d'entrée

Etape 2 : Construction des intervalles des fonctions f et g

Celle-ci concerne la construction du vecteur flou $\tilde{f}(\cdot)$ et la matrice floue $\tilde{g}(\cdot)$. Nous commençons par le traitement du vecteur des fonctions $f(\cdot) = [f_1(\cdot), f_2(\cdot), \dots, f_n(\cdot)]^t$. Nous définissons pour chaque combinaison de \tilde{x} les intervalles $I_{f_i}^{(k_1, \dots, k_n)}$ $i = 1, \dots, m$ de la manière suivante :

$$I_{f_i}^{(k_1, \dots, k_n)} = \left\{ f_i(x(t), \eta(t)) \mid x_1(t) \in I_{x_1}^{k_1}, \dots, x_n(t) \in I_{x_n}^{k_n}, \eta(t) \in P \right\} \quad (3.40)$$

Ou bien :

$$I_{f_i}^{(k_1, \dots, k_n)} = \left\{ f_i(x(t), \eta(t)) \mid x(t) \in I_x^{(k_1, \dots, k_n)}, \eta(t) \in P \right\}$$

avec : $I_x^{(k_1, \dots, k_n)} = I_{x_1}^{k_1} \times I_{x_2}^{k_2} \times \dots \times I_{x_n}^{k_n}$

Nous pouvons avoir, pour un seul élément f_i , jusqu'à $\prod_{i=1}^n M_i$ intervalles $I_{f_i}^{(k_1, \dots, k_n)}$. La largeur de chaque intervalle, correspondant à une combinaison possible du vecteur flou \tilde{x} , dépend des largeurs de $I_{x_i}^{k_i}$ et de la plage de variation des paramètres P .

L'univers de discours U_{f_i} (ou la plage de toutes les valeurs possibles) de $f_i(x(t), \eta(t))$ est défini par :

$$U_{f_i} = \bigcup_{(k_1, \dots, k_n)} I_{f_i}^{(k_1, \dots, k_n)} = \bigcup_{k_1=K_{11}}^{K_{12}} \bigcup_{k_2=K_{21}}^{K_{22}} \dots \bigcup_{k_n=K_{n1}}^{K_{n2}} I_{f_i}^{(k_1, \dots, k_n)} \quad (3.41)$$

De même, les plages des fonctions scalaires $g_{ij}(\cdot)$, $i = 1, \dots, n$; $j = 1, \dots, m$, sont déterminées de la même façon que ceux des $f_i(\cdot)$:

$$I_{g_{ij}}^{(k_1, \dots, k_n)} = \left\{ g_{ij}(x(t), \eta(t)) \mid x(t) \in I_x^{(k_1, \dots, k_n)}, \eta(t) \in P \right\} \quad (3.42)$$

pour : $k_i = K_{i1}, \dots, K_{i2}$; $i = 1, \dots, n$; $j = 1, \dots, m$.

L'univers de discours $U_{g_{ij}}$ de la fonction g_{ij} est défini par l'union des intervalles $I_{g_{ij}}^{(k_1, \dots, k_n)}$ correspondant à toutes les combinaisons possibles (k_1, \dots, k_n) du vecteur \tilde{x} :

$$U_{g_{ij}} = \bigcup_{(k_1, \dots, k_n)} I_{g_{ij}}^{(k_1, \dots, k_n)} = \bigcup_{k_1=K_{11}}^{K_{12}} \bigcup_{k_2=K_{21}}^{K_{22}} \dots \bigcup_{k_n=K_{n1}}^{K_{n2}} I_{g_{ij}}^{(k_1, \dots, k_n)} \quad (3.43)$$

Remarque :

Les deux dernières propositions garantissent que les intervalles des fonctions $I_{f_i}^{(k_1, \dots, k_n)}$ et $I_{g_{ij}}^{(k_1, \dots, k_n)}$ sont bornés quel que soit la plage des incertitudes $\eta \in P$ et pour tout $x_i \in I_{x_i}^{(k_1, \dots, k_n)}$; $i = 1, \dots, n$.

Etape 3 : Définition des ensembles flous des sorties

Dans cette étape, à chaque intervalle $I_{f_i}^{(k_1, \dots, k_n)}$ est associée une fonction d'appartenance $\mu_{f_i}^{(k_1, \dots, k_n)}$ dont la forme est choisie par le concepteur, afin de bien définir les ensembles flous $F_{f_i}^{(k_1, \dots, k_n)}$ relatifs aux fonctions f_i (Figure 3.14).

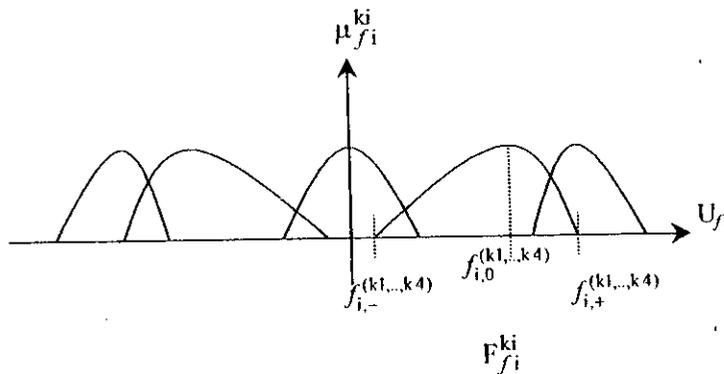


Figure 3.14 : Répartition des ensembles flous pour une fonction de sortie

La partition floue de l'univers de discours U_{f_i} est définie par :

$$P_F(U_{f_i}) = \left\{ F_{f_i}^{(k_1, \dots, k_n)} \mid k_\ell = K_{\ell_1}, \dots, K_{\ell_2} ; \ell = 1, \dots, n \right\} \quad (3.44)$$

De la même manière, nous attribuons à chaque intervalle $I_{g_{ij}}^{(k_1, \dots, k_n)}$ une fonction d'appartenance $\mu_{g_{ij}}^{(k_1, \dots, k_n)}$ afin de construire les ensembles flous $F_{g_{ij}}^{(k_1, \dots, k_n)}$ relatifs aux fonctions g_{ij} . Nous obtenons ainsi les $n \times m$ partitions floues des univers $U_{g_{ij}}$:

$$P_F(U_{g_{ij}}) = \left\{ F_{g_{ij}}^{(k_1, \dots, k_n)} \mid k_\ell = K_{\ell_1}, \dots, K_{\ell_2} ; \ell = 1, \dots, n \right\} \quad (3.45)$$

pour : $i = 1, \dots, n$ et $j = 1, \dots, m$.

Etape 4 : Déduction de la base de règles

Les fonctions floues \tilde{f}_i et \tilde{g}_i du nouveau modèle flou peuvent être concrétisées en utilisant des règles de type If-Then.

Le vecteur et la matrice de fonctions floues $\tilde{f}(\cdot)$ et $\tilde{g}(\cdot)$ sont définis respectivement par :

$$\begin{aligned} \tilde{f}(\cdot) : P_F(U_{x_1}) \times \dots \times P_F(U_{x_n}) &\rightarrow [P_F(\mathfrak{R})]^n \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{f}(\tilde{x}) \end{aligned} \quad (3.46)$$

avec : $\tilde{f}(\tilde{x}) = (F_{f_1}^{(k_1, \dots, k_n)}, \dots, F_{f_n}^{(k_1, \dots, k_n)})^t$ un vecteur flou

$F_{f_i}^{(k_1, \dots, k_n)} \in P_F(U_{f_i})$ est un ensemble flou

$k_i = K_{i1}, \dots, K_{i2} ; i = 1, \dots, n$.

de même :

$$\begin{aligned} \tilde{g}(\cdot) : P_F(U_{x_1}) \times \dots \times P_F(U_{x_n}) &\rightarrow [P_F(\mathfrak{R})]^{n \times m} \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{g}(\tilde{x}) \end{aligned} \quad (3.47)$$

avec : $\tilde{g}(\tilde{x}) = \left[F_{g_{ij}}^{(k_1, \dots, k_n)} \right]_{\substack{i=1, \dots, n \\ j=1, \dots, m}} = \begin{bmatrix} F_{g_{11}}^{(k_1, \dots, k_n)} & \dots & F_{g_{1m}}^{(k_1, \dots, k_n)} \\ \vdots & & \vdots \\ F_{g_{n1}}^{(k_1, \dots, k_n)} & \dots & F_{g_{nm}}^{(k_1, \dots, k_n)} \end{bmatrix}$ une matrice floue

$F_{g_{ij}}^{(k_1, \dots, k_n)} \in P_F(U_{g_{ij}})$

$k_i = K_{i1}, \dots, K_{i2} ; i = 1, \dots, n ; j = 1, \dots, m$.

Pour chaque fonction f_i et g_{ij} , sont définies $\prod_{i=1}^n M_i$ règles comme suit :

$$\begin{aligned} \mathfrak{R}_{f_i}^{(k_1, \dots, k_n)} : \quad \text{IF} \quad x_1 \text{ is } F_{x_1}^{k_1} \text{ AND } \dots \text{ AND } x_n \text{ is } F_{x_n}^{k_n} \\ \text{THEN} \quad f_i \text{ is } F_{f_i}^{(k_1, \dots, k_n)} \end{aligned} \quad (3.48)$$

$$\text{ou } \mathfrak{R}_{f_i}^{(k_1, \dots, k_n)} : \text{ IF } x \text{ is } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t$$

$$\text{ THEN } f_i \text{ is } F_{f_i}^{(k_1, \dots, k_n)} \quad (3.49)$$

Et de même :

$$\mathfrak{R}_{g_{ij}}^{(k_1, \dots, k_n)} : \text{ IF } x_1 \text{ is } F_{x_1}^{k_1} \text{ AND } \dots \text{ AND } x_n \text{ is } F_{x_n}^{k_n}$$

$$\text{ THEN } g_{ij} \text{ is } F_{g_{ij}}^{(k_1, \dots, k_n)} \quad (3.50)$$

D'une manière plus générale, les règles correspondant aux vecteur et matrice de fonctions $f(\cdot)$ et $g(\cdot)$ sont représentées par :

$$\mathfrak{R}_f^{(k_1, \dots, k_n)} : \text{ IF } x \text{ est } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t$$

$$\text{ THEN } f(x, \eta) \text{ est } F_f^{(k_1, \dots, k_n)} = [F_{f_1}^{(k_1, \dots, k_n)}, F_{f_2}^{(k_1, \dots, k_n)}, \dots, F_{f_n}^{(k_1, \dots, k_n)}]^t \quad (3.51)$$

avec : $k_i = K_{i1}, \dots, K_{i2}$; $i = 1, \dots, n$.

De même:

$$\mathfrak{R}_g^{(k_1, \dots, k_n)} : \text{ IF } x \text{ est } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t$$

$$\text{ THEN } g(x, \eta) \text{ est } F_g^{(k_1, \dots, k_n)} = \left[F_{g_{ij}}^{(k_1, \dots, k_n)} \right]_{\substack{i=1, \dots, n \\ j=1, \dots, m}} \quad (3.52)$$

avec : $k_i = K_{i1}, \dots, K_{i2}$; $i = 1, \dots, n$; $j = 1, \dots, m$.

$f(x, \eta)$ et $g(x, \eta)$ sont respectivement le vecteur et la matrice de fonctions caractérisant le système.

III.5.3. Évaluation de la base de règles du vecteur de commande

En se basant sur le vecteur et la matrice de fonctions floues, qui caractérisent le modèle flou du système (Figure 3.1), on peut générer l'entrée floue \tilde{u} et l'écrire sous forme d'un vecteur de fonctions floues :

$$\begin{aligned} \tilde{u}(\cdot) : P_F(U_{x1}) \times \dots \times P_F(U_{xn}) &\rightarrow [P_F(\mathfrak{R})]^m \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{u}(\tilde{x}) \end{aligned} \quad (3.53)$$

avec: $\tilde{u}(\tilde{x}) = (F_{u_1}^{(k_1, \dots, k_n)}, \dots, F_{u_n}^{(k_1, \dots, k_n)})^t$ vecteur flou de commande ;

$$k_i = K_{i1}, \dots, K_{i2}; \quad i = 1, \dots, n.$$

La base de règles du vecteur d'entrée de commande u sera alors donnée par les $\prod_{i=1}^n M_i$ règles suivantes :

$$\begin{aligned} \mathfrak{R}_u^{(k_1, \dots, k_n)} : \quad \text{IF} \quad x \text{ is } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \\ \text{THEN} \quad u \text{ is } F_u^{(k_1, \dots, k_n)} = [F_{u_1}^{(k_1, \dots, k_n)}, F_{u_2}^{(k_1, \dots, k_n)}, \dots, F_{u_n}^{(k_1, \dots, k_n)}]^t \end{aligned} \quad (3.54)$$

avec : $k_i = K_{i1}, \dots, K_{i2}; \quad i = 1, \dots, n.$

III.5.4. Obtention du modèle flou

A partir des équations (3.34), (3.51), (3.52) et (3.54), le modèle flou peut être représenté par les $\prod_{i=1}^n M_i$ règles suivantes :

$$\begin{aligned} \mathfrak{R}_{\dot{x}}^{(k_1, \dots, k_n)} : \quad \text{IF} \quad x \text{ is } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \\ \text{THEN} \quad \dot{x} \text{ is } F_{\dot{x}}^{(k_1, \dots, k_n)} = [F_{\dot{x}_1}^{(k_1, \dots, k_n)}, F_{\dot{x}_2}^{(k_1, \dots, k_n)}, \dots, F_{\dot{x}_n}^{(k_1, \dots, k_n)}]^t \end{aligned} \quad (3.55)$$

avec :

$$F_{\dot{x}_i}^{(k_1, \dots, k_n)} = F_{f_i}^{(k_1, \dots, k_n)} \tilde{+} \sum_{j=1}^m F_{g_{ij}}^{(k_1, \dots, k_n)} \tilde{-} F_{u_j}^{(k_1, \dots, k_n)} \quad (3.56)$$

où : $\tilde{\sum}$ opérateur de sommation étendue.

En conclusion, le modèle proposé suivant les quatre étapes précédemment définies est élaboré en "off-line".

III.6. Conclusion

Dans ce chapitre, nous avons présenté d'une manière générale l'aspect théorique de la modélisation floue. Les systèmes d'inférence floue, et plus particulièrement les modèles flous multivariés, peuvent être classifiés selon le partitionnement de l'espace d'entrée. Trois types de partitions ont été mentionnés : la partition homogène, la partition obtenue par concaténation d'ensembles flous d'une partition homogène, et la partition optimale. Cette dernière assure une taille plus réduite de la base de règles.

D'autre part, on peut avoir une classification selon le type de la conséquence. Quatre classes principales de modèles flous sont distinguées : le modèle linguistique (comme cas particulier le modèle de Mamdani), le modèle relationnel, le modèle de Takagi-Sugeno et le modèle de Tsukamoto.

La méthode d'élaboration de modèles flous proposée exploite la connaissance analytique du modèle du système. Elle présente l'avantage d'être plus systématique, et valable pour tout l'univers de discours considéré de l'espace d'entrée. La prise en charge de la plage des incertitudes donne une meilleure robustesse lors de l'élaboration d'une commande floue robuste. Une telle stratégie de commande est abordée dans le chapitre suivant.

*Chapitre IV***Commande Floue
Basée sur un Modèle Flou****IV.1. Introduction**

Un contrôleur flou est un système d'inférence floue ayant pour sortie des actions de commande, alors que ses entrées sont le plus fréquemment considérées comme étant l'erreur et la variation de l'erreur entre une consigne et une variable du processus à contrôler. Cette dernière représente en général la (ou les) grandeurs à réguler et d'autres grandeurs mesurées qui traduisent l'évolution du système.

Il existe plusieurs types de contrôleurs flous qui diffèrent les uns des autres par la nature des conséquences des règles, à savoir: linguistique (Mamdani,...), relationnel, Takagi-Sugeno, Tsukamoto, ...etc. L'élaboration de ces contrôleurs repose généralement sur les connaissances de l'expert ou sur des essais expérimentaux, sans avoir recours aux équations représentatives (ou au modèle) du système. Ce qui se traduit par une certaine difficulté à prévoir le comportement dynamique du processus de commande, en particulier du point de vue de la stabilité et de la robustesse. De ce fait, il est préférable, en cas de présence du modèle, de concevoir le contrôleur flou, en se basant directement soit sur le modèle du système à contrôler, soit sur des connaissances préalables du système permettant d'élaborer un modèle flou qui l'approxime.

Typiquement, dans le cas d'une commande floue basée sur un modèle flou, on peut distinguer quatre blocs principaux [Gha96c], à savoir (Figure 4.18) :

- ☛ un modèle flou,
- ☛ un contrôleur flou,
- ☛ le système à contrôler,
- ☛ un bloc de défuzzification.

Le modèle flou décrit grossièrement le comportement du système à contrôler ; il est représenté par l'une des quatre représentations floues que nous avons présentées précédemment (§III.3).

L'entrée floue de commande \tilde{u} est générée par le contrôleur flou. La phase de défuzzification permet de convertir celle-ci en une entrée réelle de commande. La méthode de défuzzification doit vérifier la stabilité asymptotique et la robustesse de la boucle de retour pour toute la plage d'incertitudes [Gha96c].

Le bloc du modèle flou, dans ce cas, fournit une base de règles et des ensembles flous nécessaires pour la conception du contrôleur flou. En pratique, ce bloc est fictif, et une seule base de règles est utilisée. Tous les calculs dans l'élaboration de la commande floue se font en "off-line".

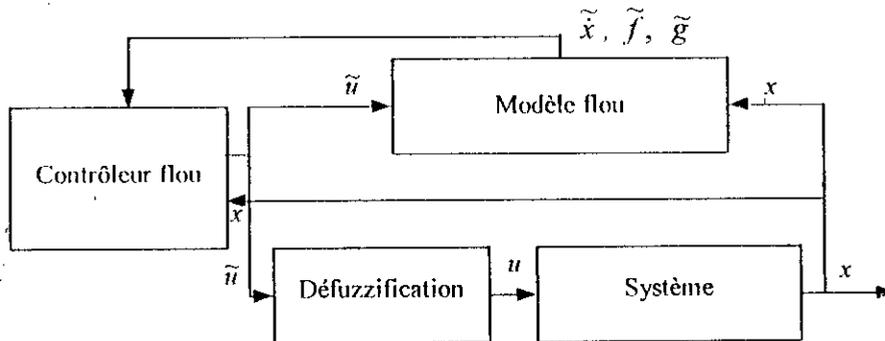


Figure 4.1 : Structure de base d'un système de commande basé sur un modèle flou

Plusieurs stratégies de commande ont échoué dans le réglage de certains systèmes, pour la simple raison qu'elles nécessitent des connaissances sur les incertitudes telles que leurs moyennes, leurs variances ou leurs types (bruit blanc, ou ayant une densité gaussienne, etc...).

La technique de modélisation floue proposée par Ben-Ghali, et que nous avons adoptée, permet de concevoir des contrôleurs flous robustes pour les systèmes incertains, sans avoir à connaître la structure ou la forme des incertitudes.

Dans ce chapitre, nous essayons de tirer profit de la modélisation floue et de l'utiliser dans deux stratégies de commande. Nous commencerons, dans le prochain paragraphe, par l'utilisation des concepts de la modélisation floue pour concevoir un modèle inverse flou, afin de l'introduire dans le cas de la commande à flux orienté du moteur asynchrone. Dans le dernier paragraphe, nous présenterons une commande floue robuste basée sur un modèle flou du système, sa validité est vérifiée dans le cas d'une commande non-linéaire du moteur asynchrone en tenant compte du modèle défini par les équations (1.32) et (1.33).

IV.2. Commande par un modèle inverse flou

Le modèle inverse est élaboré en "off-line", celui-ci est utilisé pour aider un opérateur humain à prendre une décision appropriée pour le contrôle d'un procédé. D'autre part, certains systèmes non-linéaires peuvent être directement contrôlés par leurs modèles inverses [Bab98a][Eco86].

L'élaboration d'un modèle inverse consiste à calculer une (ou m) variables d'entrée en fonction de la (ou les m) sorties du système et le reste des variables d'entrée (Figure 4.2).

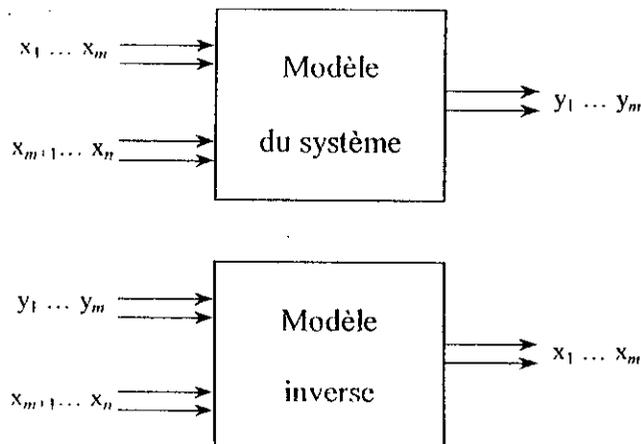


Figure 4.2 : Le modèle inverse d'un système MIMO

Plusieurs travaux ont été consacrés à l'élaboration d'un modèle inverse flou. Ce dernier est obtenu par plusieurs méthodes, à savoir :

- ☛ à partir des données expérimentales des entrées/sorties du système [Tle99][Che00][Bra96b],
- ☛ en modélisant, d'abord, le système en introduisant la logique floue et en inversant ensuite la base de règles appropriée [Bab98a],
- ☛ directement à partir d'une représentation analytique du modèle inverse, tout en tenant compte des incertitudes du système [Sel99][Gha96a][Ped97].

Dans tous les cas, on doit avoir une bijection entre les variables d'entrée et de sortie. Dans la deuxième méthode, la bijection dans la base de règles du modèle d'origine est vérifiée en testant les relations entre les noyaux des ensembles flous des sorties et ceux des entrées correspondantes, quand les entrées restantes sont des termes flous constants.

IV.2.1. Élaboration du modèle inverse flou par l'inversion d'une base de règles

Cette méthode consiste à intervertir entre les ensembles flous des variables de l'antécédent et ceux de la conséquence, tout en s'assurant que la bijection est effective.

Afin de mieux comprendre ce procédé, nous présentons un exemple d'inversion de la base de règles d'un modèle flou ayant deux entrées x_1 et x_2 et une sortie y . Le modèle inverse flou convoité a comme sortie uniquement la variable x_1 , comme le montre la figure 4.3. Néanmoins, il est important de remarquer que la table de règles qui en résulte se trouve être incomplète, ce qui représente un inconvénient majeur.

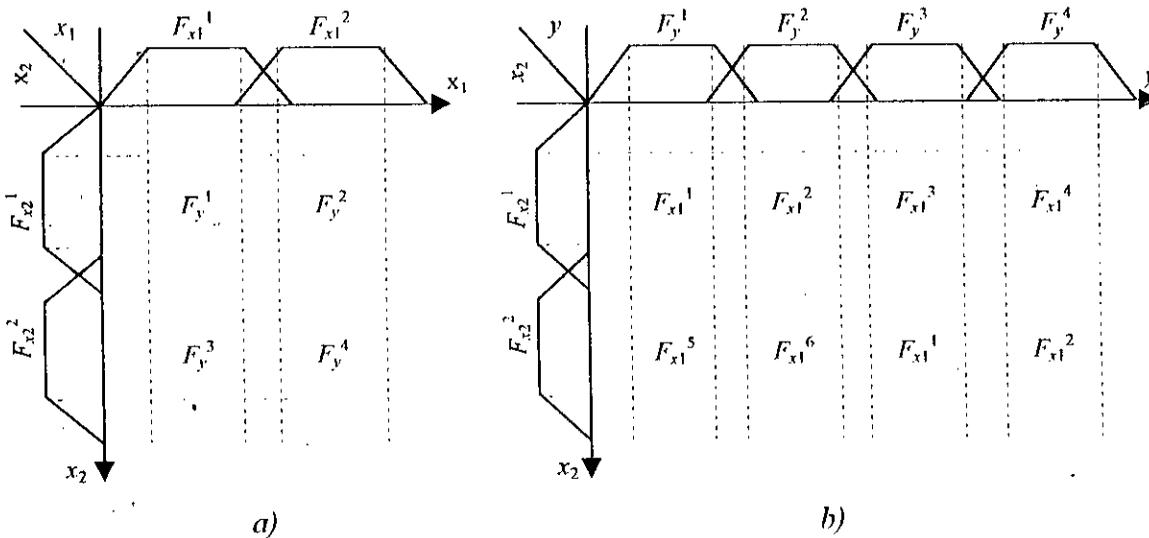


Figure 4.3 : Exemple d'inversion d'une base de règles
a) modèle flou original ; b) modèle inverse flou

Afin de pallier à ce problème, nous utilisons une des techniques présentée par Babuska [Bab98a], qui consiste à représenter au préalable géométriquement le modèle flou dans un espace à trois dimensions comme le montre la figure 4.4a. Pour ce faire, des points représentatifs (moyennes des noyaux, centre de gravité...) des ensembles flous de l'antécédent et de la conséquence sont considérés. L'ensemble de ces points représente les bornes limites de la surface correspondant au modèle flou original.

Par la suite, le modèle inverse flou peut être représenté par une nouvelle surface similaire à celle du modèle flou original à laquelle une rotation convenable des axes est effectuée (Figure 4.4b). Cette surface est également limitée par les univers de discours des variables d'entrée et de sortie choisies.

De plus, si on suppose que les ensembles flous $F_{x_1}^i$, $F_{x_2}^j$, F_y^i considérés sont représentés par les valeurs réelles x_{1i} , x_{2j} , y_i respectivement (avec $i : 1, \dots, 4$; $j : 1, 2$), les tables de

règles du système flou et de son inverse peuvent être représentées par les tables (4.1a) et (4.1b) suivantes :

$x_2 \setminus x_1$	x_{11}	x_{12}
x_{21}	$a : y_1$	$b : y_2$
x_{22}	$c : y_3$	$d : y_4$

a)

$x_2 \setminus y$	y_1	y_2	y_3	y_4
x_{21}	$a : x_{11}$	$b : x_{12}$	$e : x_{13}$	$f : x_{14}$
x_{22}	$g : x_{15}$	$h : x_{16}$	$c : x_{11}$	$d : x_{12}$

b)

Table 4.1 : Tables de règles - a) du modèle flou et b) du modèle inverse flou

La projection des points (y_3, x_{21}) et (y_2, x_{22}) du plan (y, x_2) selon l'axe des x_1 sur l'hyperplan caractérisant le système inverse flou, permet l'obtention des points e et h , respectivement. Ces derniers se situent à l'intérieur de l'hypersurface limitée par $(a, b, c$ et $d)$. Les ensembles flous correspondants à x_{13} et x_{16} sont donc à l'intérieur de l'univers de discours.

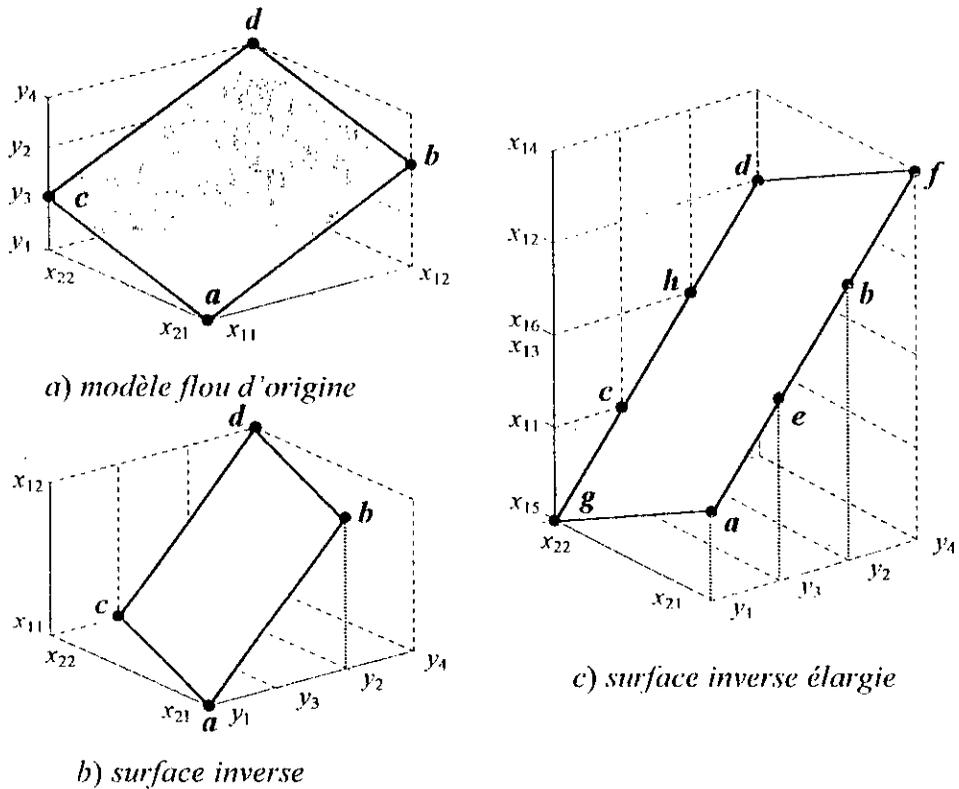


Figure 4.4 : Inversion d'une base de règle de deux entrées et une sortie

Par contre, la projection de (y_1, x_{22}) et (y_2, x_{22}) donne deux points en dehors de la surface. Il faut donc étendre la surface jusqu'aux points d'intersection g et f , ayant x_{15} et x_{16} comme coordonnées sur l'axe x_1 (Figure 4.4c). Les ensembles flous $F_{x_1}^i, F_{x_2}^j$ seront construits selon ces deux dernières valeurs. L'extension de la surface permet d'élargir l'univers de discours de la variable x_1 et donc, d'extrapoler le modèle inverse flou.

L'élaboration du modèle inverse flou par cette méthode devient de plus en plus difficile, compte tenu de l'augmentation du nombre de variables d'entrée/sortie et de la diminution de la taille de la base de règles du modèle d'origine. En plus, il s'avère être plus difficile, si on utilise une structure plus réduite de partitionnement de l'espace d'entrée.

IV.2.2. Elaboration du modèle inverse flou à partir d'une connaissance analytique du modèle inverse

Cette technique, que nous avons retenue, permet l'élaboration du modèle inverse flou en exploitant la description analytique d'un modèle inverse et la plage d'incertitudes.

Rappelons qu'un modèle inverse peut être écrit sous la forme :

$$y = f(x) \quad (4.1)$$

$y = (y_1, \dots, y_m)^t$: vecteur de commande.

$x = (x_1, \dots, x_n)^t$: vecteur d'entrée.

$f = (f_1, \dots, f_m)^t$: vecteur de fonctions caractérisant le modèle du système.

Les éléments du vecteur x sont les variables d'état du système et/ou leurs dérivées et/ou même les sorties du système.

En présence des incertitudes, l'équation (4.1) devient :

$$y = f(x, \eta) \quad (4.2)$$

où η : représente les incertitudes ($\eta \in P$), qui peuvent être paramétriques, non-paramétriques ou structurelles.

Le problème posé consiste à développer une représentation du système basée sur la logique floue. Cette représentation peut être décrite par :

$$\tilde{y} = \begin{pmatrix} \tilde{y}_1 \\ \dots \\ \tilde{y}_m \end{pmatrix} = \tilde{f}(\tilde{x}) = \begin{pmatrix} \tilde{f}_1(\tilde{x}_1, \dots, \tilde{x}_n) \\ \dots \\ \tilde{f}_m(\tilde{x}_1, \dots, \tilde{x}_n) \end{pmatrix} \quad (4.3)$$

\tilde{x} et \tilde{y} : vecteurs flous correspondant aux vecteurs x et y .

\tilde{f} : vecteur de fonctions floues.

IV.2.3. Application au moteur asynchrone

Nous avons vu précédemment que la commande vectorielle n'est autre que le modèle inverse du moteur asynchrone, selon une orientation du flux sur l'axe direct du référentiel. La modélisation floue de la commande par flux orienté (CFO) nous permet donc d'obtenir le modèle inverse flou du moteur en supposant avoir la même condition d'orientation du flux.

L'expression de la CFO développée dans le deuxième chapitre (2.2) est écrite comme suit :

$$y = f(x, \eta) \quad (4.4)$$

où $y = (\nu_{ds}, \nu_{qs}, \omega_s)^t$; $x = (\Phi_r^*, \dot{\Phi}_r^*, C_e, \omega_m)^t$;

$f = (f_1, f_2, f_3)^t$ tel que :

$$\begin{aligned} f_1(x) &= \nu_{ds} = \frac{1}{L_m} (T_r \dot{\Phi}_r^* + \Phi_r^*) \\ f_2(x) &= \nu_{qs} = \frac{L_r}{p \cdot L_m} \cdot \frac{C_e^*}{\Phi_r^*} \\ f_3(x) &= \omega_s = \omega_m + \frac{L_m}{T_r} \cdot \frac{\nu_{qs}}{\Phi_r^*} = \omega_m + \frac{L_r}{p \cdot T_r} \cdot \frac{C_e^*}{\Phi_r^{*2}} \end{aligned} \quad (4.5)$$

Il reste cependant à concevoir la représentation floue de l'équation (4.4) qui rassemble toutes les incertitudes inhérentes au système ; celle-ci est définie par :

$$\tilde{y} = \begin{pmatrix} \tilde{\nu}_{ds} \\ \tilde{\nu}_{qs} \\ \tilde{\omega}_s \end{pmatrix} = \tilde{f}(\tilde{x}) = \begin{pmatrix} \tilde{f}_1(\tilde{\Phi}_r, \tilde{\dot{\Phi}}_r) \\ \tilde{f}_2(\tilde{\Phi}_r, \tilde{C}_e) \\ \tilde{f}_3(\tilde{\Phi}_r, \tilde{C}_e, \tilde{\omega}_m) \end{pmatrix} \quad (4.6)$$

$\tilde{\Phi}_r, \tilde{\dot{\Phi}}_r, \tilde{C}_e, \tilde{\omega}_m$ et $\tilde{\nu}_{ds}, \tilde{\nu}_{qs}, \tilde{\omega}_s$: variables floues correspondant aux variables d'entrée $\Phi_r^*, \dot{\Phi}_r^*, C_e^*, \omega_m$ et aux variables de commande $\nu_{ds}, \nu_{qs}, \omega_s$ respectivement.

On considère que le flux de référence Φ_r^* est constant, ce qui signifie que $\dot{\Phi}_r^* = 0$.

Le modèle inverse flou est construit selon les quatre étapes de l'algorithme de la modélisation floue (Paragraphe 3.5). Dans ce cas, le nombre d'ensembles flous des variables de sortie $\nu_{ds}, \nu_{qs}, \omega_s$ est égal à $M_1, M_1 \times M_3$ et $M_1 \times M_3 \times M_4$ respectivement. Les règles sont présentées comme suit :

$$\begin{aligned} \mathfrak{R}_{\nu_{ds}}^{(k_1, \dots, k_4)} : \quad & \text{IF } \Phi_r \text{ is } F_{\Phi_r}^{k_1} \\ & \text{THEN } \nu_{ds} \text{ is } F_{\nu_{ds}}^{(k_1, \dots, k_4)} = F_{\nu_{ds}}^{k_1} \end{aligned} \quad (4.7)$$

$$\mathfrak{R}_{\iota_{qs}}^{(k_1, \dots, k_4)} : \text{ IF } \Phi_r \text{ is } F_{\Phi_r}^{k_1} \text{ AND } C_e \text{ est } F_{C_e}^{k_3} \text{ THEN } \iota_{qs} \text{ is } F_{\iota_{qs}}^{(k_1, k_3)} \tag{4.8}$$

$$\mathfrak{R}_{\omega_s}^{(k_1, \dots, k_4)} : \text{ IF } (\Phi_r, C_e, \omega_m) \text{ is } (F_{\Phi_r}^{k_1}, F_{C_e}^{k_3}, F_{\omega_m}^{k_4}) \text{ THEN } \omega_s \text{ is } F_{\omega_s}^{(k_1, k_3, k_4)} \tag{4.9}$$

Les entrées Φ_r^* , ω_m sont soumises à une fuzzification afin de les convertir en données floues. Par contre, l'entrée C_e peut ne pas être fuzzifiée si on utilise un régulateur flou précédant le bloc CFO.

La défuzzification est calculée suivant la méthode de la moyenne pondérée.

$$f_i = \frac{\sum_{k_1} \sum_{k_3} \sum_{k_4} \tilde{f}_i^{(k_1, k_3, k_4)} \cdot T(\mu_{x_1}^{k_1}, \mu_{x_3}^{k_3}, \mu_{x_4}^{k_4})}{\sum_{k_1} \sum_{k_3} \sum_{k_4} T(\mu_{x_1}^{k_1}, \mu_{x_3}^{k_3}, \mu_{x_4}^{k_4})} \tag{4.10}$$

f_i représenté une des variables de sortie, $i = 1, \dots, 3$.

$\tilde{f}_i^{(k_1, k_3, k_4)}$ est la valeur représentative de l'ensemble flou $F_{f_i}^{(k_1, k_3, k_4)}$ qui est la sortie de la règle $R_{f_i}^{(k_1, k_3, k_4)}$.

T : opération de T-norme, qui renvoie dans notre cas le produit de ses arguments.

Il reste à présent au concepteur de faire le choix du nombre, de la répartition et de la forme des ensembles flous, pour à chaque variable d'entrée C_e^* , Φ_r^* et ω_m , ainsi que le choix de la forme des ensembles flous des variables de sortie ι_{ds} , ι_{qs} , ω_s , et de définir les plages de variation des paramètres L_m , L_r , et R_r .

Le schéma bloc correspondant au modèle inverse flou ainsi conçu est représenté par la figure 4.5.

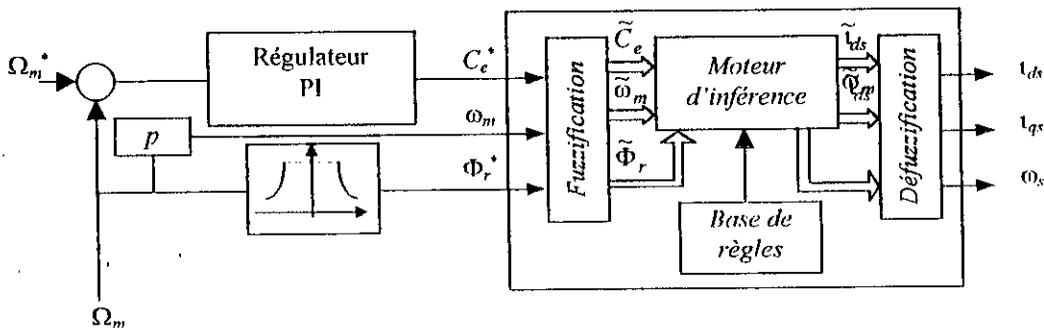


Figure 4.5 : Schéma de commande par un modèle inverse flou remplaçant la CFO

Théoriquement, nous pouvons remarquer que plus le nombre de sous-ensembles flous des entrées augmente, plus l'approximation est meilleure. Néanmoins, on ne peut dépasser un certain seuil, au risque de perdre les avantages apportés par la logique floue, entre autres: l'augmentation du temps de calcul.

Par conséquent, il serait judicieux de réduire la complexité du modèle inverse flou. Dans ce sens, le moyen le plus simple consiste à subdiviser le modèle en plusieurs blocs (Figure 4.6) [Bab98b]. Si ceci s'avérait insuffisant, d'autres méthodes de réduction pourraient être utilisées, à savoir : la méthode RFS (Referential Fuzzy Sets) [Ped84], l'interpolation [Kóc97], ou les mesures de similarité dans les bases de règles floues [Set95][Set98a][Set98b].

IV.2.4. Décomposition du modèle inverse flou

Afin de réduire la complexité du modèle inverse flou proposé, nous avons décomposé ce dernier en quatre sous-blocs moins complexes que le précédent. Chaque bloc ayant au plus deux entrées, comme le montre la figure 4.6.

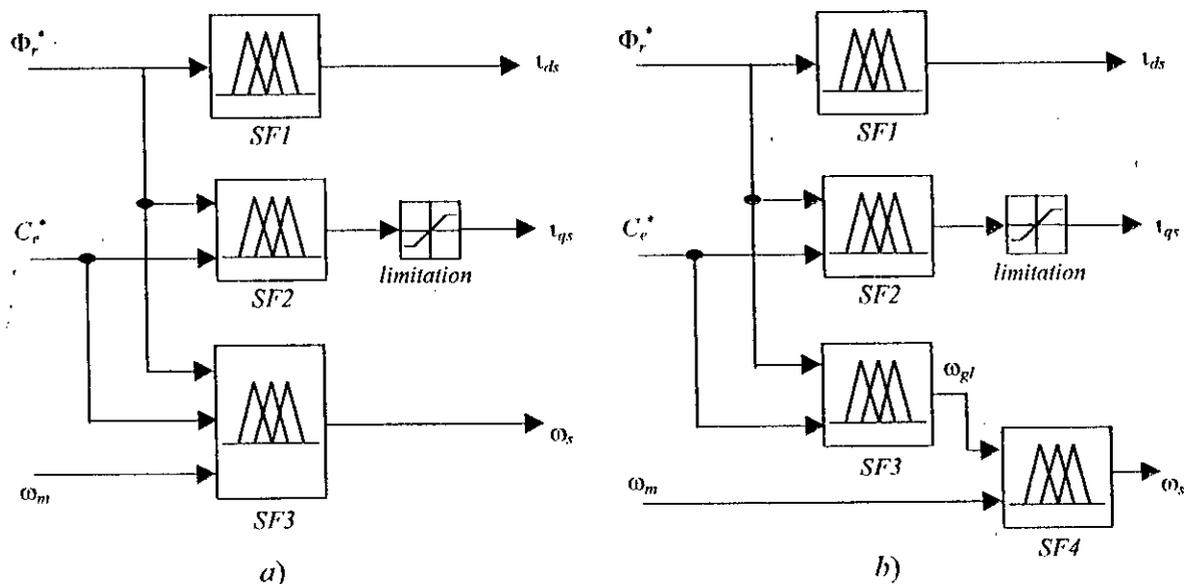


Figure 4.6 : Décomposition du modèle inverse flou en des sous-systèmes flous

Comme le quatrième sous-bloc *SF4* (Figure 4.6a) représente en réalité une sommation entre les deux variables ω_{gl} et ω_m , il serait judicieux, lors de l'implémentation, de le remplacer par un sommateur (Figure 4.6b).

IV.2.5. Résultats de simulation

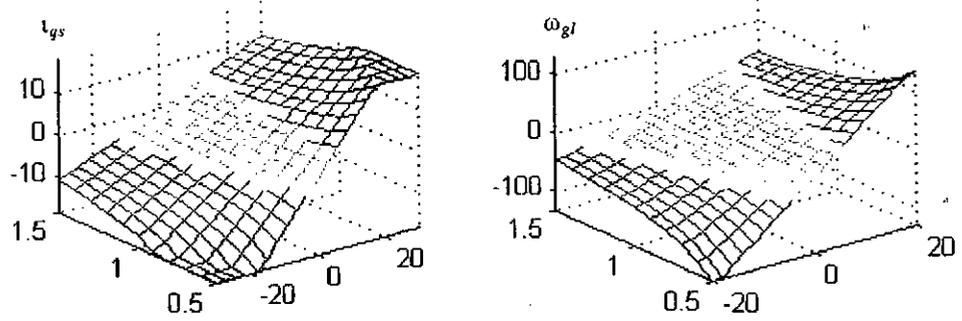
Une simulation numérique a été effectuée sur le système utilisant le modèle inverse flou de la figure 4.6b. L'essai est similaire à celui utilisé au paragraphe §II.2 qui consiste à faire démarrer le moteur à vide, avec une consigne de vitesse de 100 rad/sec, un flux rotorique de référence de 1 Wb, un couple résistant de 10 Nm à l'instant $t = 0.5$ sec, et une inversion de vitesse de 100 rad/sec à -100 rad/sec à l'instant $t = 2$ sec.

La simulation a été faite hors de la zone de défluxage (d'où $\dot{\Phi}_r^* = 0$) et sans saturation, en utilisant un régulateur PI classique. Les formes des fonctions d'appartenance adoptées pour les variables d'entrée et de sortie du modèle inverse flou sont de type triangulaire. De plus, afin de montrer l'impact de ces valeurs sur les performances dynamiques du système, ce même essai a été effectué avec différents nombres de fonctions d'appartenance M_i ($i = 1, 2, 3$) en prenant successivement les valeurs 3, 5, 7, 9, 11 et 15.

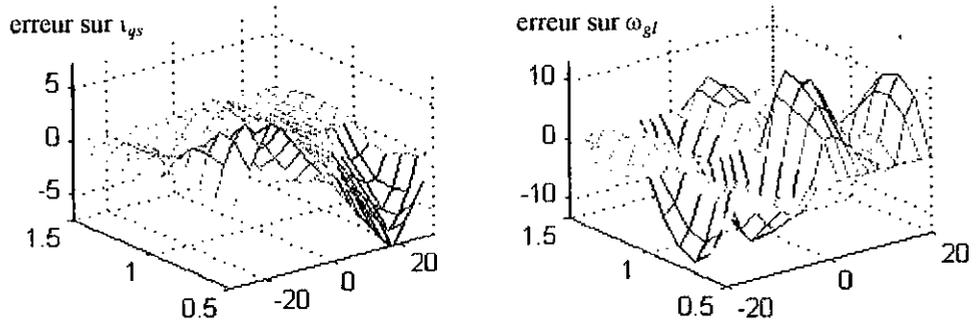
Les résultats de simulation représentés par les figures 4.7 et 4.8 montrent clairement que le modèle inverse flou adopté a pu décrire le modèle inverse classique avec un certain degré de précision. L'erreur entre les deux commandes est d'autant plus faible qu'on augmente le nombre de partitions des univers de discours des variables d'entrée.

Nous avons également effectué un test de robustesse par rapport aux variations paramétriques pour un nombre de partitions égal à 7. La résistance rotorique a été variée de 90% à 150% par rapport à sa valeur nominale, alors qu'une incertitude de 5% sur l'inductance rotorique et l'inductance mutuelle a été imposée (Figure 4.9).

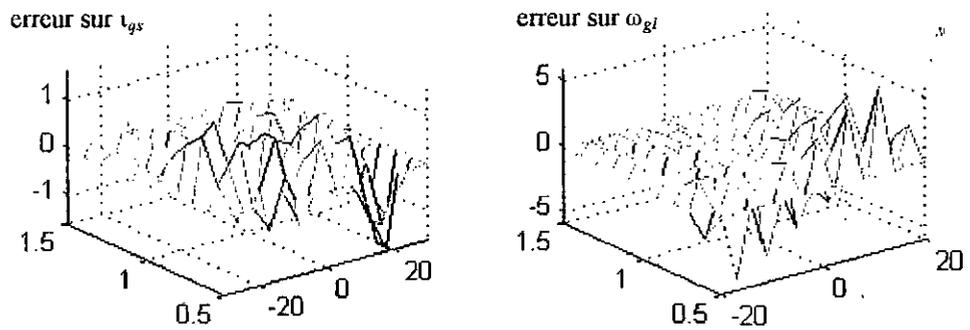
Les résultats de simulation montrent que les réponses dynamiques dans ce cas ont les mêmes allures que celles obtenues avec la CFO classique. Ce qui nous permet de dire que le modèle inverse flou adopté a pu décrire le modèle inverse classique. Le modèle flou garde donc les inconvénients de la commande classique (la sensibilité vis-à-vis des paramètres). Néanmoins, son avantage réside dans le fait qu'il permet une implémentation facile, une rapidité dans les calculs, et une facilité d'ajustage après la mise en œuvre de la commande dans la pratique, ce qui n'est pas le cas du modèle inverse classique.



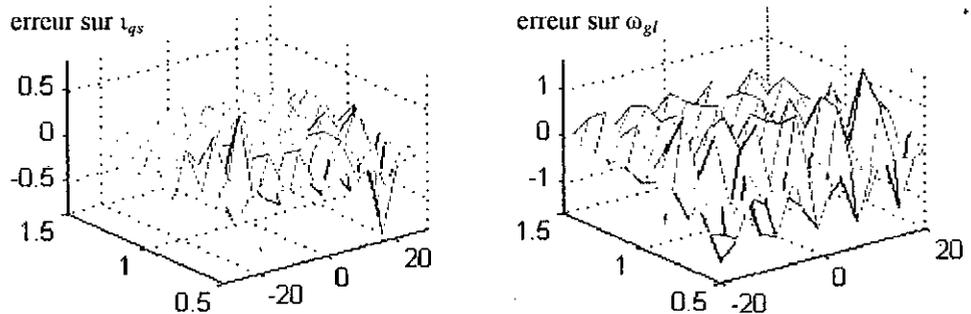
a) sorties de commande pour le cas réel



b) erreur entre la commande réelle et floue pour $M_i = 3$



c) erreur entre la commande réelle et floue pour $M_i = 7$



d) erreur entre la commande réelle et floue pour $M_i = 15$

Figure 4.7 : Commande par un modèle inverse flou. Comparaison entre les deux commandes réelle et floue pour différentes valeurs de M_i

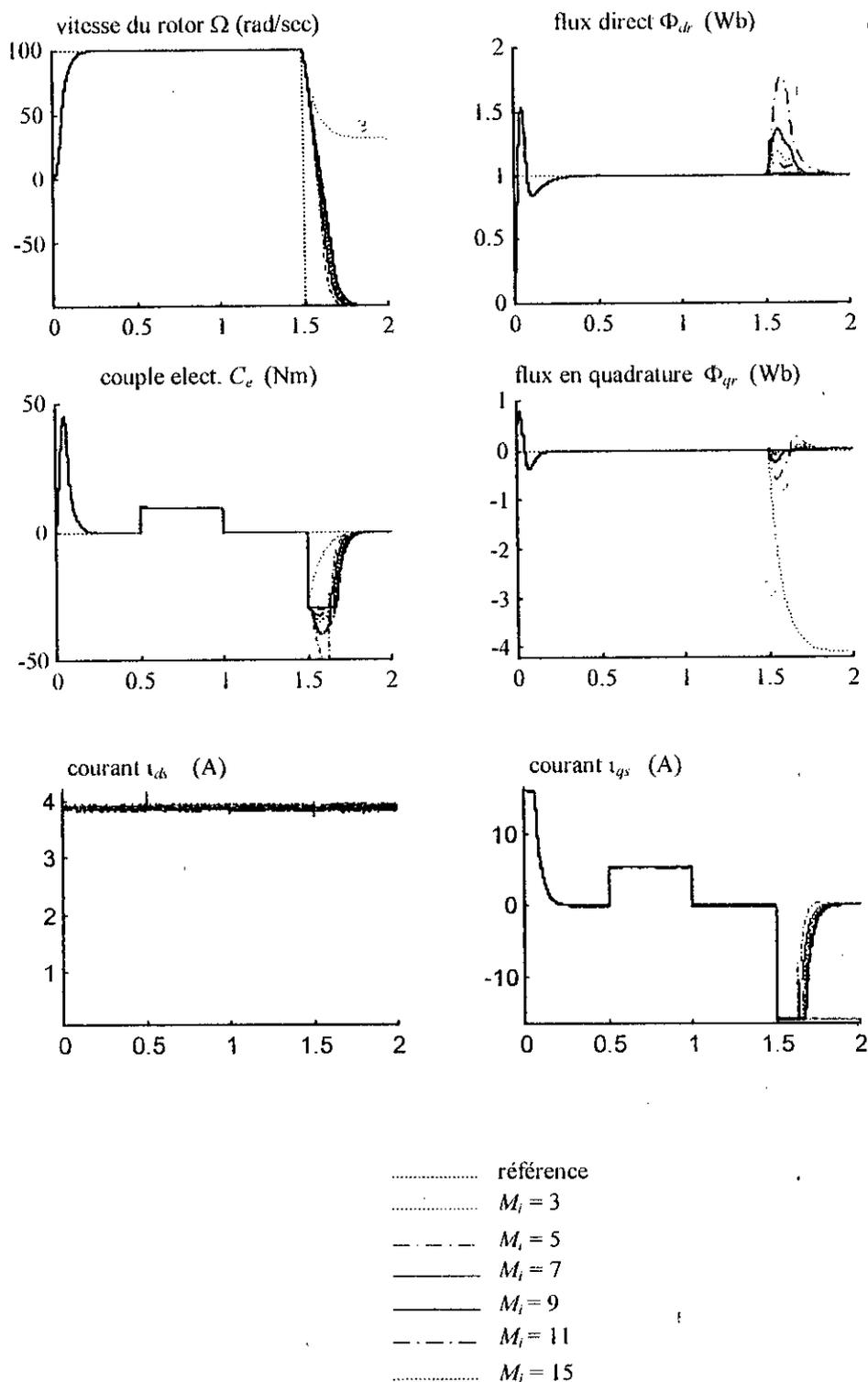


Figure 4.8 : Commande par un modèle inverse flou. Résultats de simulation pour différentes valeurs de M_i (sans variation des paramètres)

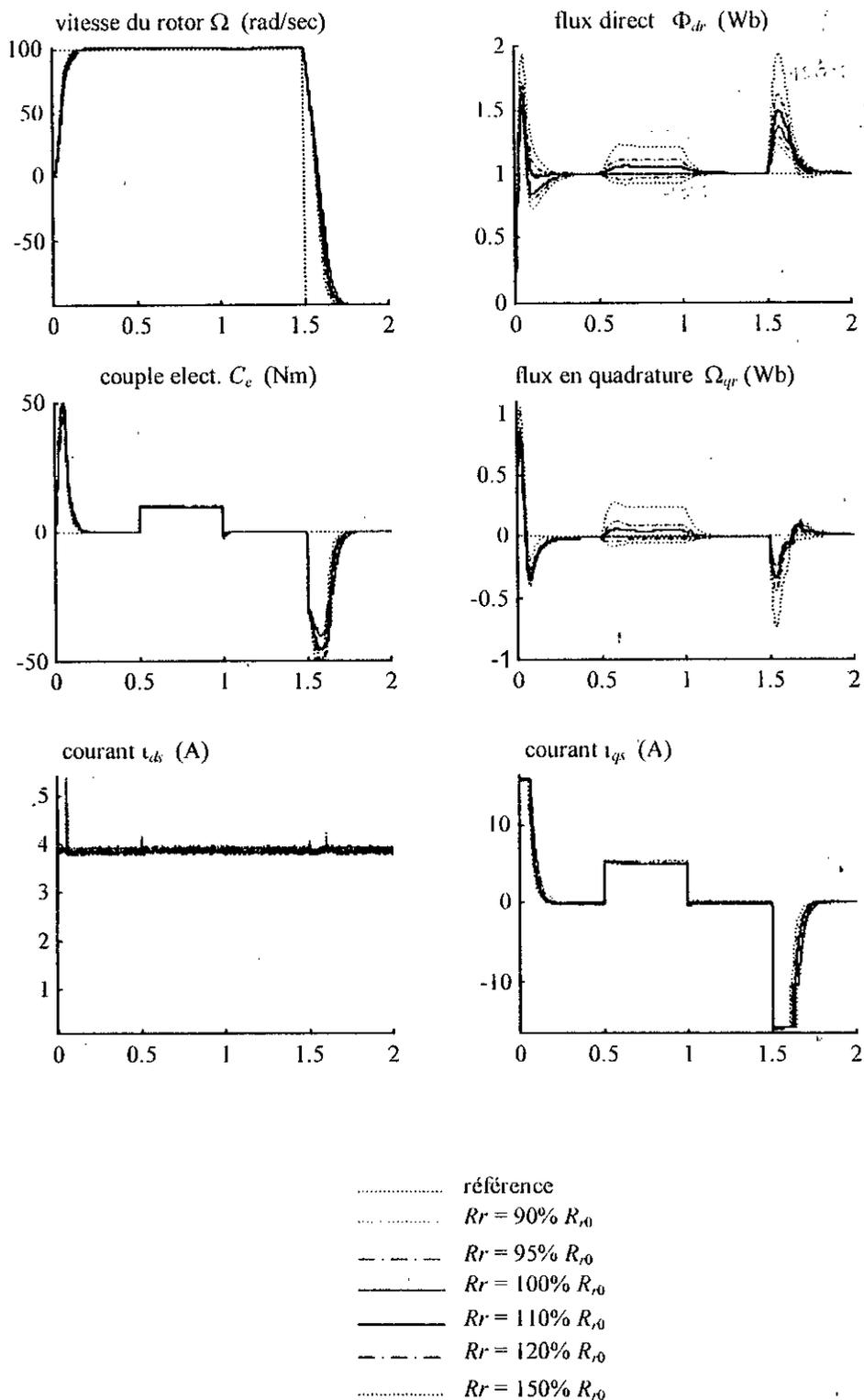


Figure 4.9 : Commande par un modèle inverse flou. Résultats de simulation pour différentes variations de R_r (avec un partitionnement de l'espace d'entrée de 7 ensembles flous et une variation de 5% sur les inductances)

IV.3. Commande à Structure Variable Floue

L'approche proposée pour la conception d'un contrôleur flou est une extension des techniques de commande à structures variables (CSV) pour les systèmes flous représentés par des équations différentielles floues.

Notons qu'il existe des travaux qui intègrent la logique floue dans la CSV (Chapitre 2) [Dri96][He97][Mid00][Bou99b][Ish93][Hsu98]. Cependant, la CSV floue proposée est différente des études précédentes, dans le sens où elle est basée sur l'exploitation du modèle flou du système [Gha96b][Gha97].

Dans l'approche proposée, le contrôleur flou (Figure 4.1) calcule l'entrée de commande floue \tilde{u} qui est le résultat du modèle flou décrit du système [Gha97]. L'entrée réelle de commande sera calculée en défuzzifiant \tilde{u} .

Soit le système décrit par l'équation (3.33) et par sa représentation floue suivante :

$$\tilde{\dot{x}} = \tilde{f}(\tilde{x}) \tilde{+} \tilde{g}(\tilde{x}) \tilde{\sim} \tilde{u} \quad (4.11)$$

avec :

$$\tilde{f}(x) = \begin{pmatrix} \tilde{f}_1(\tilde{x}_1, \dots, \tilde{x}_n) \\ \dots \\ \tilde{f}_1(\tilde{x}_1, \dots, \tilde{x}_n) \end{pmatrix} ; \quad \tilde{g}(\tilde{x}) = [g_{ij}(\tilde{x}_1, \dots, \tilde{x}_n)]_{\substack{i=1, \dots, n \\ j=1, \dots, m}} ; \quad \tilde{u}(t) = \begin{pmatrix} \tilde{u}_1(t) \\ \dots \\ \tilde{u}_m(t) \end{pmatrix}$$

Avant d'expliquer l'algorithme de la CSV floue, nous rappelons quelques notations et définitions utilisées par ce dernier.

Soit $U_x = U_{x_1} \times \dots \times U_{x_n}$ et $U_u = U_{u_1} \times \dots \times U_{u_m}$ les univers de discours (ou l'espace) des vecteurs d'entrée et de commande respectivement.

L'univers de discours du vecteur d'erreur e est défini par :

$$U_e = \left\{ e \in \mathbb{R}^n / e = x_d - x ; x \in U_x \right\} \quad (4.12)$$

Le vecteur d'erreur flou, $\tilde{e}(t) \in P_F(U_e)$, correspondant au vecteur $e(t)$ est donné par :

$$\tilde{e}(t) = \tilde{x}_d(t) \tilde{\sim} \tilde{x}(t) = \begin{pmatrix} \tilde{e}_1(t) \\ \dots \\ \tilde{e}_n(t) \end{pmatrix} = \begin{pmatrix} \tilde{x}_{d1}(t) - \tilde{x}_1(t) \\ \dots \\ \tilde{x}_{dn}(t) - \tilde{x}_n(t) \end{pmatrix} \quad (4.13)$$

avec $\tilde{e}_i(t) = \tilde{x}_{di}(t) \tilde{\sim} \tilde{x}_i(t)$ pour $i = 1, \dots, n$;

et $\tilde{x}_{di}(t)$ un singleton flou centré sur la $i^{\text{ème}}$ référence x_{di} .

L'univers de discours du vecteur de commutation $s(x)$, défini par l'équation (2.6), est décrit par :

$$U_s = U_{s_1} \times U_{s_2} \times \dots \times U_{s_m} \subset \mathbb{R}^m \quad (4.14)$$

$$U_{s_i} = \{ s_i(x) \in \mathbb{R} \mid e \in U_e \} \quad ; \quad i = 1, \dots, m$$

U_{s_i} est l'ensemble de toutes les variables possibles de la fonction de commutation s_i pour tout $e \in U_e$.

Définition 4.1 : Le vecteur flou de commutation \tilde{s} est un vecteur de m applications, défini par :

$$\begin{aligned} \tilde{s} : P_F(U_e) &\rightarrow P_F(U_s) \\ \tilde{e} &\rightarrow \tilde{s}(\tilde{e}) \end{aligned} \quad (4.15)$$

avec $\tilde{s}(\tilde{e})$ l'extension de (2.6) :

$$\tilde{s}(\tilde{e}) = \begin{pmatrix} \tilde{s}_1(\tilde{e}) \\ \dots \\ \tilde{s}_m(\tilde{e}) \end{pmatrix} \quad (4.16)$$

$\tilde{s}_i(\tilde{e}) \in P_F(U_{s_i})$ la fonction floue de commutation définie dans U_{s_i} et qui correspond au vecteur d'erreur \tilde{e} défini dans U_e .

Définition 4.2 : La surface floue de glissement est définie par :

$$\begin{aligned} \tilde{S}_s &= \tilde{S}_{s_1} \times \dots \times \tilde{S}_{s_m} \\ \tilde{S}_{s_i} &= \left\{ \tilde{e} \in P_F(U_e) \mid \tilde{s}_i(\tilde{e}) = \tilde{0} \right\} \end{aligned} \quad (4.17)$$

avec : $\tilde{0}$ un ensemble flou constant entourant la valeur 0.

L'étendue de la dynamique désirée \hat{s}_d peut être définie par le vecteur flou :

$$\tilde{\hat{s}}(\tilde{e}) = \begin{pmatrix} \tilde{\hat{s}}_1(\tilde{e}) \\ \dots \\ \tilde{\hat{s}}_m(\tilde{e}) \end{pmatrix} \in P_F(U_{\hat{s}}) \quad (4.18)$$

avec $P_F(U_{\hat{s}}) = P_F(U_{\hat{s}_1}) \times \dots \times P_F(U_{\hat{s}_m})$

Définition 4.3 : La condition d'atteignabilité est définie par :

$$\exists \tilde{N} \text{ tel que } \tilde{s}(\tilde{e})' \cdot \tilde{\hat{s}}(\tilde{e}) \subset \tilde{N} \quad (4.19)$$

\tilde{N} est un ensemble flou dont le support est strictement négatif.

Lemme 4.1 : La condition d'atteignabilité (4.19) est équivalente à la condition (4.20) telle que :

$$\forall s_i(e) \in \text{Supp}(\tilde{s}_i(\tilde{e})) - \{0\} \quad \text{et} \quad \forall \hat{s}_i(e) \in \text{Supp}(\tilde{\hat{s}}_i(\tilde{e})) - \{0\} \quad (4.20)$$

$$\text{Alors : } s_i(e) \cdot \hat{s}_i(e) < 0$$

Preuve : (voir Annexe C) [Gha97].

IV.3.1. Algorithme de l'élaboration de la CSV floue

L'algorithme de l'élaboration de la commande à structure variable floue, basée sur le modèle flou linguistique présenté dans le chapitre précédent, peut être réalisé en quatre étapes :

Étape 1. Construction du vecteur flou d'erreur $F_e^{(k_1, \dots, k_n)}$

Soit $P_F(U_x)$ et $P_F(U_e)$ les partitions floues des univers de discours U_x et U_e respectivement, telles que :

$$P_F(U_x) = P_F(U_{x_1}) \times \dots \times P_F(U_{x_n}) \quad (4.21)$$

$$P_F(U_e) = P_F(U_{e_1}) \times \dots \times P_F(U_{e_n})$$

avec : $P_F(U_{e_i}) = \left\{ F_{e_i}^{k_i} \text{ défini dans } U_{e_i} \text{ de } \mathbb{R} ; k_i = 1, \dots, M_i ; i = 1, \dots, n \right\}$

$F_{e_i}^{k_i} = \tilde{x}_{di} \simeq F_{x_i}^{k_i}$ des ensembles flous que peut prendre la variable floue \tilde{e}_i .

Le vecteur d'erreur $e \in \mathbb{R}^n$ est caractérisé par les $\prod_{i=1}^n M_i$ vecteurs flous :

$$F_e^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{e_1}^{k_1} \\ \dots \\ F_{e_n}^{k_n} \end{pmatrix} ; k_i = 1, \dots, M_i ; i = 1, \dots, n \quad (4.22)$$

Cependant le vecteur flou \tilde{e} prend un des $\prod_{i=1}^n M_i$ vecteurs flous $F_e^{(k_1, \dots, k_n)}$.

Étape 2. Construction du vecteur flou de commutation $F_s^{(k_1, \dots, k_n)}$

Soit $P_F(U_s)$ la partition floue de l'univers de discours U_s :

$$P_F(U_s) = P_F(U_{s_1}) \times \dots \times P_F(U_{s_m}) \quad (4.23)$$

avec $P_F(U_s) = \left\{ F_s^{(k_1, \dots, k_n)} \text{ défini dans } U_s \text{ de } \mathbb{R}^m ; k_i = 1, \dots, M_i ; i = 1, \dots, n \right\}$

et $P_F(U_{s_j}) = \left\{ F_{s_j}^{(k_1, \dots, k_n)} \text{ défini dans } U_{s_j} \text{ de } \mathbb{R} ; k_j = 1, \dots, M_j ; j = 1, \dots, m \right\}$

Le vecteur flou $F_s^{(k_1, \dots, k_n)}$ est constitué des m ensembles flous $F_{s_i}^{(k_1, \dots, k_n)}$ définissant les fonctions floues de commutation $\tilde{s}_i(\tilde{e})$, tel que :

$$F_s^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{s_1}^{(k_1, \dots, k_n)} \\ \dots \\ F_{s_m}^{(k_1, \dots, k_n)} \end{pmatrix} \quad (4.24)$$

avec : $F_{s_i}^{(k_1, \dots, k_n)} = \{s_i(e) \in U_{s_i} \mid e \in U_e\}$

La base de règles correspondante à s_i est présentée par les $\prod_{i=1}^n M_i$ règles de type If-Then :

$$\begin{aligned} \mathfrak{R}_{s_i}^{(k_1, \dots, k_n)} : \quad & \text{IF} \quad e = (e_1, \dots, e_n)^t \quad \text{is} \quad F_e^{(k_1, \dots, k_n)} = (F_{e_1}^{k_1}, \dots, F_{e_n}^{k_n})^t \\ & \text{THEN} \quad s_i(e) \quad \text{is} \quad F_{s_i}^{(k_1, \dots, k_n)} \end{aligned} \quad (4.25)$$

Étape 3. Choix de la dynamique floue désirée $F_s^{(k_1, \dots, k_n)}$

Les fonctions \hat{s}_i , désignant la dynamique désirée du système, sont fonction des signes de s_i , de telle sorte que la condition d'atteignabilité régie par l'équation (2.9) soit vérifiée.

Dans le cas d'une commande à structure variable floue, le problème se pose sur l'ensemble flou $F_{s_i}^{(k_1, \dots, k_n)}$, $i = 1, \dots, n$; dont le support contient une partie positive, et une autre négative de l'univers de discours U_{s_i} (Figure 4.10). Dans ce cas, l'ensemble flou $F_{s_i}^{(k_1, \dots, k_n)}$ est décomposé en deux parties : $PF_{s_i}^{(k_1, \dots, k_n)}$ et $NF_{s_i}^{(k_1, \dots, k_n)}$ qui sont des ensembles flous positif et négatif respectivement. L'ensemble $F_{s_i}^{(k_1, \dots, k_n)}$ vérifie les conditions suivantes :

$$\begin{aligned} \text{supp}(PF_{s_i}^{(k_1, \dots, k_n)}) \cap \text{supp}(NF_{s_i}^{(k_1, \dots, k_n)}) &= \emptyset \\ \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) &= \text{supp}(PF_{s_i}^{(k_1, \dots, k_n)}) \cup \text{supp}(NF_{s_i}^{(k_1, \dots, k_n)}) \cup \{0\} \\ \mu(PF_{s_i}^{(k_1, \dots, k_n)}) &= \mu(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si} \quad s > 0 \quad \text{et} \quad 0 \text{ ailleurs} \\ \mu(NF_{s_i}^{(k_1, \dots, k_n)}) &= \mu(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si} \quad s < 0 \quad \text{et} \quad 0 \text{ ailleurs} \end{aligned} \quad (4.26)$$

où $\text{supp}(F)$ représente le support de l'ensemble flou F .

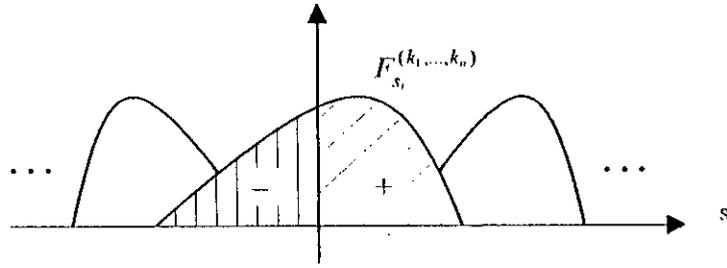


Figure 4.10 : Décomposition des ensembles flous en des ensembles flous positifs et négatifs

On définit le vecteur flou \tilde{s}_d caractérisé par l'un des $\prod_{i=1}^n M_i$ vecteurs d'ensembles flous $F_{\tilde{s}_i}^{(k_1, \dots, k_n)}$ tel que :

$$F_{\tilde{s}}^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{\tilde{s}_1}^{(k_1, \dots, k_n)} \\ \dots \\ F_{\tilde{s}_m}^{(k_1, \dots, k_n)} \end{pmatrix} \quad (4.27)$$

Le vecteur flou \tilde{s}_d définit grossièrement le comportement du système en dehors de la surface de glissement.

Chaque terme $F_{\tilde{s}_i}^{(k_1, \dots, k_n)}$ commute entre deux ensembles flous suivant le signe de la fonction de commutation s , et doit garantir la condition floue d'atteignabilité (4.19).

$F_{\tilde{s}_i}^{(k_1, \dots, k_n)}$ peut être défini par :

$$\begin{aligned} F_{\tilde{s}_i}^{(k_1, \dots, k_n)} &= -\tilde{M}(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si } \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \subset U_{s_i}^+ \\ &= +\tilde{M}(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si } \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \subset U_{s_i}^- \end{aligned} \quad (4.28)$$

et par :

$$F_{\tilde{s}_i}^{(k_1, \dots, k_n)} = -\omega_s \tilde{M}(PF_{s_i}^{(k_1, \dots, k_n)}) \tilde{\mp} (1 - \omega_s) \tilde{M}(NF_{s_i}^{(k_1, \dots, k_n)}) \quad (4.29)$$

si $\text{supp}(F_{s_i}^{(k_1, \dots, k_n)})$ contient en même temps une partie négative et une autre positive de U_{s_i}

avec :

$$\begin{aligned} \omega_s &= 1 \quad \text{si } s_i \in \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \cap U_{s_i}^+ \\ &= 0 \quad \text{si } s_i \in \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \cap U_{s_i}^- \end{aligned} \quad (4.30)$$

$U_{s_i}^+ \subset \mathbb{R}^+$ et $U_{s_i}^- \subset \mathbb{R}^-$ sont les parties strictement positive et strictement négative respectivement de $U_{s_i} \subset \mathbb{R}$.

La base de règles qui caractérise la dynamique floue désirée, durant la phase transitoire d'atteignabilité, consiste en $\prod_{i=1}^m M_i$ règles suivantes :

$$\begin{aligned} \mathfrak{R}_{\dot{s}_i}^{(k_1, \dots, k_n)} : \quad & \text{IF} \quad e = (e_1, \dots, e_n)^t \quad \text{is} \quad F_e^{(k_1, \dots, k_n)} = (F_{e_1}^{k_1}, \dots, F_{e_n}^{k_n})^t \\ & \text{THEN} \quad \dot{s}_i(e) \quad \text{is} \quad F_{\dot{s}_i}^{(k_1, \dots, k_n)} \end{aligned} \quad (4.31)$$

$$i = 1, \dots, m ; k_j = 1, \dots, M_j ; j = 1, \dots, n$$

Étape 4. Calcul de la loi de commande floue $F_u^{(k_1, \dots, k_n)}$

Les fonctions de commutation doivent avoir un degré relatif égale à 1, c'est-à-dire que les entrées de commandes u_i doivent apparaître dès la première dérivée de s_i . Cela permet l'extraction de la loi de commande à partir de la dérivée \dot{s} .

L'extension de l'équation (2.16) permet l'obtention de la loi de commande floue qui sera définie par :

$$\tilde{u} = \tilde{m} \vee \left\{ (\tilde{s}_x(\tilde{e}))^t \sim \tilde{g}(\tilde{x}) \right\} \sim \left\{ \tilde{s}_d(\tilde{e}) \approx (\tilde{s}_x)^t \sim \tilde{f}(\tilde{x}) \approx \tilde{s}_t \right\} \quad (4.32)$$

avec :

$$\tilde{s}_x \text{ et } \tilde{s}_t \text{ des vecteurs flous obtenus par l'extension des vecteurs } \frac{\partial s}{\partial x} \text{ et } \frac{\partial s}{\partial t} \text{ respecti-}$$

vement.

$\tilde{f}(\tilde{x})$ et $\tilde{g}(\tilde{x})$ sont respectivement le vecteur et la matrice flous caractérisant le modèle flou du système.

$\tilde{u} = (\tilde{u}_1, \dots, \tilde{u}_m)^t$ le vecteur flou de commande qui est l'un des vecteurs d'ensembles flous :

$$F_u^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{u_1}^{(k_1, \dots, k_n)} \\ \dots \\ F_{u_m}^{(k_1, \dots, k_n)} \end{pmatrix} \quad (4.33)$$

La base de règles correspondante à la loi de commande sera donc :

$$\begin{aligned} \mathfrak{R}_{u_i}^{(k_1, \dots, k_n)} : \\ \text{IF} \quad x = (x_1, \dots, x_n)^t \quad \text{is} \quad F_x^{(k_1, \dots, k_n)} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n})^t \quad \text{AND} \quad s_i \quad \text{is} \quad F_{s_i}^{(k_1, \dots, k_n)} \\ \text{THEN} \quad u_i \quad \text{is} \quad F_{u_i}^{(k_1, \dots, k_n)} \end{aligned} \quad (4.34)$$

$$i = 1, \dots, m ; k_j = 1, \dots, M_j ; j = 1, \dots, n$$

Défuzzification de la loi de commande floue

Chaque règle produit les ensembles flous $F_{u_i}^{(k_1, \dots, k_n)}$, qui doivent être défuzzifiés afin de fournir une valeur réelle pour chaque sortie de commande. La défuzzification peut être obtenue par la méthode de la moyenne pondérée, définie par :

$$u_i = \frac{\sum_{k_1} \dots \sum_{k_n} \bar{u}_i^{(k_1, \dots, k_n)} \cdot T(\mu_{x_1}^{k_1}, \dots, \mu_{x_n}^{k_n})}{\sum_{k_1} \dots \sum_{k_n} T(\mu_{x_1}^{k_1}, \dots, \mu_{x_n}^{k_n})} \tag{4.35}$$

u_i représente l'une des variables de sortie, $i = 1, \dots, m$.

$\bar{u}_i^{(k_1, \dots, k_n)}$ est la valeur représentative de l'ensemble flou $F_{u_i}^{(k_1, \dots, k_n)}$ qui est la sortie de la règle $R_{u_i}^{(k_1, \dots, k_n)}$; $i = 1, \dots, m$.

T : opération de T-norme, qui renvoie dans notre cas le produit de ses arguments.

IV.3.2. Cas des systèmes mis sous forme canonique

Dans notre cas, le modèle de la machine asynchrone est mis sous plusieurs blocs, chacun représente un sous-système mono-entrée / mono-sortie décrit par l'équation (2.18).

Puisque les blocs sont distincts, et pour simplifier les notations, posons :

$$z = x^i \quad ; \quad f(\cdot) = f^i(\cdot) \quad ; \quad g(\cdot) = g^i(\cdot) \quad ; \quad u = u_i \quad ; \quad r = r_i$$

Chaque sous-système est alors caractérisé par l'équation différentielle suivante :

$$\dot{z} = f(x) + g(x) \cdot u \tag{4.36}$$

avec :

$$z = \begin{pmatrix} x_1^i \\ \vdots \\ x_r^i \end{pmatrix} = \begin{pmatrix} z_1 \\ \vdots \\ z_r \end{pmatrix} \quad ; \quad f(x) = \begin{pmatrix} f_1(x) \\ \vdots \\ f_{r-1}(x) \\ f_r(x) \end{pmatrix} = \begin{pmatrix} z_2 \\ \vdots \\ z_r \\ a(x) \end{pmatrix} \quad ; \quad g(x) = \begin{pmatrix} g_1(x) \\ \vdots \\ g_{r-1}(x) \\ g_r(x) \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ b(x) \end{pmatrix}$$

$$x \in \mathbb{R}^n \quad ; \quad z \in \mathbb{R}^r$$

n, r : les dimensions des sous-systèmes et du système global, respectivement.

$a(\cdot), b(\cdot) : \mathbb{R}^n \times P \rightarrow \mathbb{R}$ sont des fonctions scalaires caractérisant le modèle du sous-système.

IV.3.2.1. Objectif de la commande

Soit :

$$z_d = \begin{pmatrix} z_{d_1}(t) \\ z_{d_2}(t) \\ \vdots \\ z_{d_r}(t) \end{pmatrix} = \begin{pmatrix} z_{d_1} \\ \dot{z}_{d_1} \\ \vdots \\ z_{d_1}^{(r-1)} \end{pmatrix} \text{ le vecteur de référence (ou la trajectoire désirée) du sous-système.}$$

U_u : l'univers de discours de l'entrée de commande du sous-système.

$\eta \in P$: représente les incertitudes inhérentes dans le modèle.

Le problème est de trouver la commande u (de chaque sous-système) de telle sorte que, pour tout incertitude $\eta \in P$, le système global soit stable et converge asymptotiquement vers les vecteurs de référence, tout en supposant que les dynamiques internes soient stables (au moins bornées).

IV.3.2.2. Modélisation floue du sous-système

Le sous-système est décrit par la nouvelle représentation floue qui est une équation différentielle floue donnée par :

$$\tilde{\dot{z}} = \tilde{f}(\tilde{x}) \tilde{+} \tilde{g}(\tilde{x}) \tilde{\approx} \tilde{u} \quad (4.37)$$

avec :

$$\tilde{z} = \begin{pmatrix} \tilde{z}_1 \\ \vdots \\ \tilde{z}_r \end{pmatrix} ; \quad \tilde{x} = \begin{pmatrix} \tilde{x}_1 \\ \vdots \\ \tilde{x}_n \end{pmatrix} \text{ les vecteurs d'état flous ; } r \leq n$$

$$\tilde{f}(\tilde{x}) = \begin{pmatrix} \tilde{f}_1(\tilde{x}) \\ \vdots \\ \tilde{f}_{r-1}(\tilde{x}) \\ \tilde{f}_r(\tilde{x}) \end{pmatrix} = \begin{pmatrix} \tilde{z}_2 \\ \vdots \\ \tilde{z}_r \\ \tilde{a}(\tilde{x}) \end{pmatrix} ; \quad \tilde{g}(\tilde{x}) = \begin{pmatrix} \tilde{g}_1(\tilde{x}) \\ \vdots \\ \tilde{g}_{r-1}(\tilde{x}) \\ \tilde{g}_r(\tilde{x}) \end{pmatrix} = \begin{pmatrix} \tilde{0} \\ \vdots \\ \tilde{0} \\ \tilde{b}(\tilde{x}) \end{pmatrix}$$

$\tilde{0}$: est un singleton flou centré sur la valeur « zéro ».

La modélisation floue du présent modèle suit les mêmes étapes que celles de la section (§III.5) du chapitre précédent.

Après l'exécution de l'algorithme, nous obtenons les ensembles flous $F_{f_i}^{(k_1, \dots, k_n)}$, $F_{g_i}^{(k_1, \dots, k_n)}$ ($k_j = 1, \dots, M_j$; $j = 1, \dots, n$; $i = 1, \dots, r$) correspondant aux fonctions $f_i(\cdot)$ et $g_i(\cdot)$, et qui dépendent de la combinaison du vecteur d'état flou \tilde{x} et de la plage des incertitudes P .

Les fonctions $f_i(\cdot)$; $i = 1, \dots, r-1$ sont mono-variables. Les ensembles flous correspondants sont plus faciles à obtenir et prennent chacun comme valeur l'un des éléments du vecteur flou \tilde{x} .

L'extension des fonctions restantes $a(\cdot)$ et $b(\cdot)$ mène à deux applications floues $\tilde{a}(\cdot)$ et $\tilde{b}(\cdot)$ données par :

$$\begin{aligned} \tilde{a}(\cdot) : P_F(U_{x_1}) \times \dots \times P_F(U_{x_n}) &\rightarrow P_F(\mathbb{R}) \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{a}(\tilde{x}) = F_a^{(k_1, \dots, k_n)} \end{aligned} \quad (4.38)$$

$$\begin{aligned} \tilde{b}(\cdot) : P_F(U_{x_1}) \times \dots \times P_F(U_{x_n}) &\rightarrow P_F(\mathbb{R}) \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{b}(\tilde{x}) = F_b^{(k_1, \dots, k_n)} \end{aligned} \quad (4.39)$$

La base de règles correspondante se caractérise par $\prod_{i=1}^n M_i$ règles au maximum pour chaque fonction.

IV.3.2.3. Élaboration de la CSV floue

Il reste maintenant à trouver une commande floue robuste pour chaque sortie, en se basant sur le modèle flou résultant du sous-système. On procède alors aux étapes suivantes :

Étape 1. Construction du vecteur d'erreur

Cette étape est la même que celle de la section précédente. Le vecteur d'erreur est défini par :

$$e = z - z_d = \begin{pmatrix} z_1 \\ \vdots \\ z_r \end{pmatrix} - \begin{pmatrix} z_{d_1} \\ \vdots \\ z_{d_r} \end{pmatrix}$$

Le vecteur d'erreur flou sera donc :

$$\tilde{e} = \tilde{z}_d - \tilde{z} = \begin{pmatrix} \tilde{z}_{d_1} \\ \vdots \\ \tilde{z}_{d_r} \end{pmatrix} - \begin{pmatrix} \tilde{z}_1 \\ \vdots \\ \tilde{z}_r \end{pmatrix}$$

Le vecteur d'ensembles flous que peut prendre le vecteur d'erreur flou est donné par :

$$F_e^{(k_1, \dots, k_r)} = \begin{pmatrix} F_{e_1}^{k_1} \\ \vdots \\ F_{e_r}^{k_r} \end{pmatrix} \in P_F(U_e) \quad (4.40)$$

avec: $F_{e_i}^{k_i} = \tilde{z}_{d_i} \approx F_{z_i}^{k_i}$

$$k_i = 1, \dots, M_i ; i = 1, \dots, r ;$$

\approx : l'opérateur de soustraction étendue.

$P_F(U_e) = P_F(U_{e_1}) \times \dots \times P_F(U_{e_r})$ la partition floue de l'espace du vecteur e

U_{e_i} : l'univers de discours de e_i .

Étape 2. Construction de la fonction floue de commutation

Soit la partition floue de l'univers de discours de la fonction de commutation :

$$P_F(U_s) = \left\{ F_s^{(k_1, \dots, k_r)} ; k_i = 1, \dots, M_i ; i = 1, \dots, r \right\}$$

L'extension de l'équation (2.19) nous donne :

$$\tilde{s}(\tilde{e}) = \lambda_1 \sim \tilde{e}_1 \tilde{+} \lambda_2 \sim \tilde{e}_2 \tilde{+} \dots \tilde{+} \lambda_r \sim \tilde{e}_r \quad (4.41)$$

Les équations (4.40) et (4.41) permettent de construire les ensembles flous de la fonction de commutation $F_s^{(k_1, \dots, k_r)}$, tels que :

$$F_s^{(k_1, \dots, k_r)} = \lambda_1 \sim F_{e_1}^{k_1} \tilde{+} \lambda_2 \sim F_{e_2}^{k_2} \tilde{+} \dots \tilde{+} \lambda_r \sim F_{e_r}^{k_r} \quad (4.42)$$

$\tilde{s}(\tilde{e})$ représente l'un des $\prod_{i=1}^r M_i$ ensembles flous $F_s^{(k_1, \dots, k_r)}$ définis par (4.42).

Étape 3. Choix de la dynamique floue désirée $F_s^{(k_1, \dots, k_r)}$

Pendant le mode d'atteignabilité, la robustesse est assurée par le choix de la dynamique floue désirée \tilde{s}_d , qui dépend de la fonction floue de commutation $\tilde{s}(\tilde{e})$.

Les ensembles flous $F_s^{(k_1, \dots, k_r)}$ sont choisis en se référant à l'extension d'une fonction linéaire ou non-linéaire $M(s)$.

$$F_s^{(k_1, \dots, k_r)} = -\omega_s \sim F_{M(PF_s)}^{(k_1, \dots, k_r)} \tilde{+} (\omega_s - 1) \sim F_{M(NF_s)}^{(k_1, \dots, k_r)} \quad (4.43)$$

avec

$$\begin{aligned} \omega_s &= 1 & \text{si } s \in \text{supp}(F_s^{(k_1, \dots, k_r)}) \cap U_{s_i}^+ \\ \omega_s &= 0 & \text{si } s \in \text{supp}(F_s^{(k_1, \dots, k_r)}) \cap U_{s_i}^- \end{aligned} \quad (4.44)$$

$U_{s_i}^+ \subset \mathbb{R}$ et $U_{s_i}^- \subset \mathbb{R}$ sont les parties strictement positive et strictement négative de

$U_{s_i} \subset \mathbb{R}$.

$F_{M(PF_s)}^{(k_1, \dots, k_r)}$ sont les ensembles flous que peut prendre $\tilde{M}(\tilde{s})$ lorsque $\tilde{s} = PF_s^{(k_1, \dots, k_r)}$.

$F_{M(NF_s)}^{(k_1, \dots, k_r)}$ sont les ensembles flous que peut prendre $\tilde{M}(\tilde{s})$ lorsque $\tilde{s} = NF_s^{(k_1, \dots, k_r)}$.

Étape 4. Calcul de la loi de commande $F_u^{(k_1, \dots, k_r)}$

La fonction floue correspondant à $\tilde{s}(e)$ est obtenue à partir des équations (2.23) :

$$\tilde{s}(\tilde{e}) = \lambda_1 \tilde{e}_2 \tilde{+} \lambda_2 \tilde{e}_3 \tilde{+} \dots \tilde{+} \lambda_{r-1} \tilde{e}_r \tilde{+} \lambda_r \tilde{e}_r \quad (4.45)$$

avec :

$$\tilde{e}_r = \tilde{z}_{dr} \tilde{\approx} \tilde{z}_r \quad (4.46)$$

Lorsque le sous-système atteint une dynamique floue désirée $\tilde{s}_d(\tilde{e})$, choisie dans l'étape précédente, il sera forcé de suivre le modèle flou suivant :

$$\tilde{z}_r = \tilde{z}_{dr} \tilde{+} \frac{1}{\lambda_n} \tilde{e}_d(\tilde{e}) \tilde{+} \frac{\lambda_1}{\lambda_r} \tilde{e}_2 \tilde{\approx} \frac{\lambda_2}{\lambda_r} \tilde{e}_3 \tilde{\approx} \dots \tilde{\approx} \frac{\lambda_{r-1}}{\lambda_r} \tilde{e}_r \quad (4.47)$$

D'autre part, à partir de l'équation (4.37), la dynamique floue \tilde{z}_r du sous-système est décrite par :

$$\tilde{z}_r = \tilde{a}(\tilde{x}) \tilde{+} \tilde{b}(\tilde{x}) \tilde{\approx} \tilde{u} \quad (4.48)$$

Il en résulte donc, d'après les deux dernières équations, la loi de commande floue :

$$\tilde{u}_r = \left(-\tilde{a}(\tilde{x}) \tilde{+} \tilde{z}_{dr} \tilde{+} \frac{1}{\lambda_n} \tilde{e}_d(\tilde{e}) \tilde{+} \frac{\lambda_1}{\lambda_r} \tilde{e}_2 \tilde{\approx} \dots \tilde{\approx} \frac{\lambda_{r-1}}{\lambda_r} \tilde{e}_r \right) \tilde{\approx} \tilde{b}(\tilde{x}) \quad (4.49)$$

Cette équation nous permet de déduire les ensembles flous $F_u^{(k_1, \dots, k_r)}$, tels que :

$$F_u^{(k_1, \dots, k_r)} = \left(-F_a^{(k_1, \dots, k_n)} \tilde{+} \tilde{z}_{dr} \tilde{+} \frac{1}{\lambda_n} F_{\tilde{s}_d}^{(k_1, \dots, k_r)} \tilde{+} \frac{\lambda_1}{\lambda_r} F_{e_2}^{k_{e_2}} \tilde{\approx} \dots \tilde{\approx} \frac{\lambda_{r-1}}{\lambda_r} F_{e_r}^{k_{e_r}} \right) \tilde{\approx} F_b^{(k_1, \dots, k_n)} \quad (4.50)$$

et la base de règles correspondant à l'entrée de commande u pour chaque sous-système :

$$\begin{aligned} \mathfrak{R}_u^{(k_1, \dots, k_r)} : \quad & \text{IF } z = (z_1, \dots, z_r)^t \text{ is } F_z^{(k_1, \dots, k_r)} = (F_{z_1}^{k_1}, \dots, F_{z_r}^{k_r})^t \\ & \text{THEN } u \text{ is } F_u^{(k_1, \dots, k_r)} \end{aligned} \quad (4.51)$$

IV.3.3. Application à la commande du moteur asynchrone

IV.3.3.1. Introduction

Le modèle adopté du moteur asynchrone est représenté sous une forme canonique et correspond aux équations (1.32)-(1.33) du premier chapitre.

Comme il a été mentionné, le système (1.33) dépend étroitement des paramètres de la machine et du couple résistant, ce qui nous incite à l'approximer par un modèle flou afin de concevoir une commande floue robuste basée sur ce modèle, comme le montre la figure 4.11.

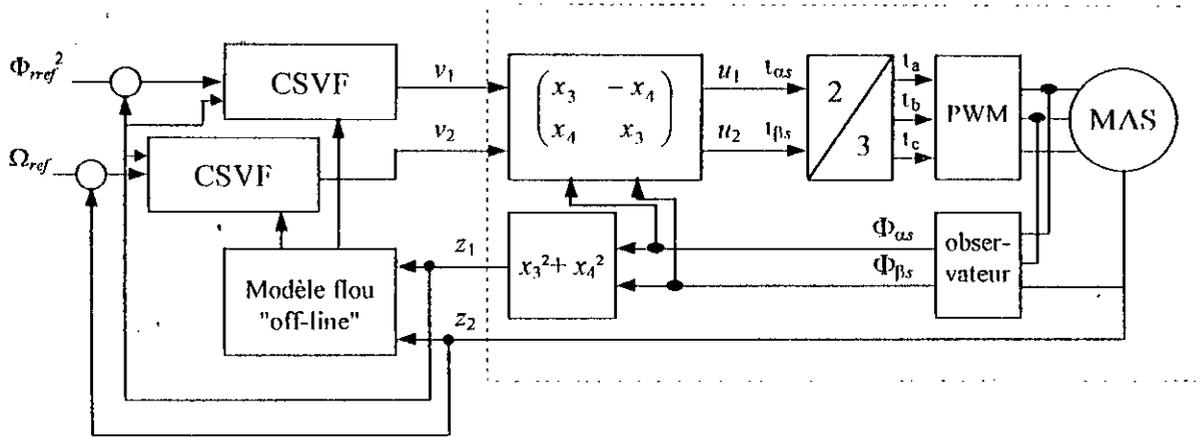


Figure 4.11 : Schéma de Commande à SV floue basée sur le modèle flou du moteur asynchrone

IV.3.3.2. Conception du modèle flou du moteur asynchrone

L'équation différentielle floue caractérisant le modèle flou du moteur, est définie par :

$$\begin{aligned}\tilde{z}_1 &= \tilde{f}_1(\tilde{z}_1) \tilde{+} \tilde{g}_1(\tilde{z}_1) \tilde{\sim} \tilde{v}_1 \\ \tilde{z}_2 &= \tilde{a} \tilde{+} \tilde{g}_2(\tilde{z}_1) \tilde{\sim} \tilde{v}_2\end{aligned}\quad (4.52)$$

où \tilde{a} est un ensemble flou constant.

avec:

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} x_3 & -x_4 \\ x_4 & x_3 \end{pmatrix} \cdot \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}\quad (4.53)$$

Algorithme de la modélisation floue :**Étape 1. Partitionnement de l'espace d'entrée**

La première étape de la modélisation floue consiste à construire une partition floue homogène de l'univers de discours U_{z_1} de la composante z_1 . C'est-à-dire définir M_1 ensembles flous $F_{z_1}^{k_1}$ correspondant à cette variable.

$I_{z_1}^{k_1} = \text{supp}(F_{z_1}^{k_1})$ représente le support du k_1 ème ensemble flou de la variable linguistique z_1 .

$$U_{z_1} \subseteq \bigcup_{k_1=1}^{M_1} I_{z_1}^{k_1} \quad (4.54)$$

Les ensembles flous $F_{z_1}^{k_1}$ ($k_1 = 1, \dots, M_1$) forment une partition floue de U_{z_1} :

$$P_F(U_{z_1}) = \{ F_{z_1}^{k_1} \mid k_1 = 1, \dots, M_1 \} \quad (4.55)$$

avec la cardinalité de $P_F(U_{z_1}) = M_1$.

La largeur de ce support, la cardinalité de la partition floue ainsi que la forme des fonctions d'appartenance sont choisies par le concepteur.

La variable \tilde{z}_1 est alors associée à l'un des M_1 ensembles flous $F_{z_1}^{k_1}$ dans l'univers de discours U_{z_1} .

Étape 2. Construction des intervalles des fonctions f_1 , g_1 et g_2

Cette étape concerne la construction des fonctions floues $\tilde{f}_1(\tilde{z}_1)$, $\tilde{g}_1(\tilde{z}_1)$ et $\tilde{g}_2(\tilde{z}_1)$. Nous commençons par le traitement de la fonction $\tilde{f}_1(\tilde{z}_1)$. Chaque valeur de la variable floue \tilde{z}_1 représente un ensemble flou $F_{z_1}^{k_1}$ de l'univers de discours U_{z_1} . Par la suite, nous définissons les intervalles $I_{f_1}^{(k_1)}$ de la fonction $f_1(\cdot)$ de la manière suivante :

$$I_{f_1}^{(k_1)} = \{ f_1(z_1(t), \eta(t)) \mid z_1(t) \in I_{z_1}^{k_1}, \eta(t) \in P \} \quad (4.56)$$

Où P représente la plage de variations de tous les paramètres de la machine et les perturbations externes.

L'univers de discours U_{f_1} (ou la plage de toutes les valeurs possibles) de $f_1(z_1(t), \eta(t))$ est défini par :

$$U_{f_1} = \bigcup_{k_1=1}^{M_1} I_{f_1}^{(k_1)} \quad (4.57)$$

On procède de la même manière pour les deux autres fonctions $g_1(z_1)$ et $g_2(z_1)$, ce qui nous permet d'aboutir à :

$$I_{g_1}^{(k_1)} = \left\{ g_1(z_1(t), \eta(t)) / z_1(t) \in I_{z_1}^{k_1}, \eta(t) \in P \right\} \quad (4.58)$$

$$U_{g_1} = \bigcup_{k_1=1}^{M_1} I_{g_1}^{(k_1)} \quad (4.59)$$

et :

$$I_{g_2}^{(k_1)} = \left\{ g_2(z_1(t), \eta(t)) / z_1(t) \in I_{z_1}^{k_1}, \eta(t) \in P \right\} \quad (4.60)$$

$$U_{g_2} = \bigcup_{k_1=1}^{M_1} I_{g_2}^{(k_1)} \quad (4.61)$$

pour : $k_1 = 1, \dots, M_1$.

Étape 3. Définition des ensembles flous des sorties

Dans cette étape, à chaque intervalle $I_{f_1}^{(k_1)}$ est associé une fonction d'appartenance $\mu_{f_1}^{(k_1)}$ afin de bien définir les ensembles flous $F_{f_1}^{(k_1)}$ relatifs aux fonctions f_1 . Dans notre cas, on a opté pour la forme triangulaire.

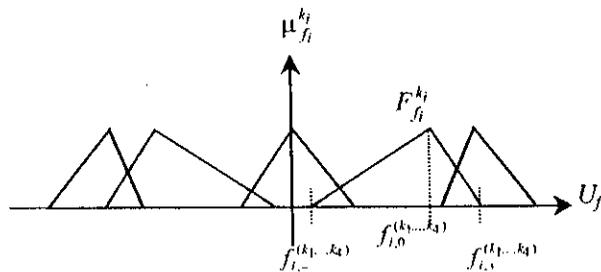


Figure 4.12 : Répartition des ensembles flous pour une fonction de sortie

La partition floue de l'univers de discours U_{f_1} est définie par :

$$P_F(U_{f_1}) = \left\{ F_{f_1}^{(k_1)} / k_1 = 1, \dots, M_1 \right\} \quad (4.62)$$

Le même procédé est suivi pour les fonctions $g_1(z_1)$ et $g_2(z_1)$.

Étape 4. Dédire la base de règles

Les fonctions floues \tilde{f}_i et \tilde{g}_i ($i = 1, 2$) du nouveau modèle flou peuvent être concrétisées en utilisant des règles de type If-Then.

Les fonctions floues \tilde{f}_i et \tilde{g}_i ($i = 1, 2$) sont définies respectivement par :

$$\begin{aligned} \tilde{f}_i : P_F(U_{z_i}) &\rightarrow P_F(\mathbb{R}) \\ \tilde{z}_i = F_{z_i}^{k_i} &\rightarrow \tilde{f}_i(\tilde{z}_i) \end{aligned} \quad (4.63)$$

$$\begin{aligned} \text{et par : } \tilde{g}_i : P_F(U_{z_i}) &\rightarrow P_F(\mathbb{R}) \\ \tilde{z}_i = F_{z_i}^{k_i} &\rightarrow \tilde{g}_i(\tilde{z}_i) \end{aligned} \quad (4.64)$$

avec : $F_{f_i}^{(k_i)} \in P_F(U_{f_i})$ est un ensemble flou que peut prendre la fonction floue $\tilde{f}_i(\tilde{z}_i)$.

$F_{g_i}^{(k_i)} \in P_F(U_{g_i})$ est un ensemble flou que peut prendre la fonction floue $\tilde{g}_i(\tilde{z}_i)$.

$k_i = 1, \dots, M_i, i = 1, 2.$

Pour chaque fonction, f_1, g_1 et g_2 , sont définies M_i règles comme suit :

$$\begin{aligned} \mathfrak{R}^{(k_i)} : \text{ IF } z_1 \text{ is } F_{z_1}^{k_i} \\ \text{ THEN } [f_1(z_1, \eta), g_1(z_1, \eta), g_2(z_1, \eta)]' \text{ is } [F_{f_1}^{(k_i)}, F_{g_1}^{(k_i)}, F_{g_2}^{(k_i)}]' \end{aligned} \quad (4.65)$$

Ceci nous permet d'avoir la représentation du système (1.33) par le modèle flou correspondant.

IV.3.3.3. Élaboration de la commande floue robuste

Une fois le modèle flou obtenu, nous pouvons passer à la conception du contrôleur flou. D'une manière générale, plusieurs contrôleurs flous peuvent être élaborés. Étant donné que le schéma de commande est à structures variables, nous avons opté pour la conception d'un contrôleur flou robuste ; pour cela, les étapes à suivre sont les suivantes [Bar00c] :

Étape 1. Construction des variables d'erreur floues

Les variables d'erreur $e_1(t)$ et $e_2(t)$ sont définies comme:

$$\begin{aligned} e_1 &= z_{ref1} - z_1 \\ e_2 &= z_{ref2} - z_2 \end{aligned} \quad (4.66)$$

z_{ref1} et z_{ref2} sont choisies constantes.

Les variables d'erreur floues \tilde{e}_1 et \tilde{e}_2 correspondantes sont :

$$\tilde{e}_1 = \tilde{z}_{ref1} - \tilde{z}_1 \quad (4.67)$$

$$\tilde{e}_2 = \tilde{z}_{ref2} - \tilde{z}_2$$

avec : \tilde{z}_{ref1} et \tilde{z}_{ref2} des singletons flous.

La partition floue $P_F(U_{e_i})$ de l'univers de discours U_{e_i} de chaque variable d'entrée ($i = 1, 2$) est construite comme suit :

$$P_F(U_{e_i}) = \left\{ F_{e_i}^{k_i} = \tilde{z}_{ref i} - F_{z_i}^{k_i} ; k_i = 1, \dots, M_i \right\} \quad (4.68)$$

où $F_{e_i}^{k_i}$, $F_{e_2}^{k_2}$ ($k_1 = 1, \dots, M_1$; $k_2 = 1, \dots, M_2$) sont des ensembles flous que peuvent prendre, dans le présent algorithme, les variables floues \tilde{e}_1 et \tilde{e}_2 .

Étape 2. Construction des fonctions floues de commutation

Les extensions floues des fonctions de commutation $s_1(e_1)$ et $s_2(e_2)$ peuvent être écrites comme suit :

$$\tilde{s}_1(\tilde{e}_1) = \lambda_1 \tilde{e}_1 \quad \text{et} \quad \tilde{s}_2(\tilde{e}_2) = \lambda_2 \tilde{e}_2 \quad (4.69)$$

Les ensembles flous que peut prendre chacune des fonctions floues de commutation sont :

$$F_{s_1}^{k_1} = \lambda_1 \tilde{e}_1 \quad (4.70)$$

$$F_{s_2}^{k_2} = \lambda_2 \tilde{e}_2$$

Étape 3. Choix de la dynamique floue désirée

Durant le mode d'atteignabilité, la robustesse est assurée par le choix des deux dynamiques désirées \tilde{s}_{d1} et \tilde{s}_{d2} . Ces dernières sont des extensions des dynamiques désirées \dot{s}_{d1} et \dot{s}_{d2} définies par l'équation (2.34).

Les ensembles flous $F_{\tilde{s}_i}^{k_i}$ ($i = 1, 2$) correspondant à \tilde{s}_{di} , sont choisis en se référant à l'extension des deux fonctions non-linéaires $M_1(s_1)$ et $M_2(s_2)$.

$$F_{\tilde{s}_i}^{(k_i)} = -\omega_{s_i} \tilde{e}_i \tilde{F}_{M(PF_{s_i})}^{(k_i)} + (\omega_{s_i} - 1) \tilde{F}_{M(NF_{s_i})}^{(k_i)} \quad (4.71)$$

avec

$$\omega_{s_i} = \begin{cases} 1 & \text{si } s_i \in \text{supp}(F_{s_i}^{k_i}) \cap U_{s_i}^+ \\ 0 & \text{si } s_i \in \text{supp}(F_{s_i}^{k_i}) \cap U_{s_i}^- \end{cases} \quad (4.72)$$

$U_{s_i}^+ \subset \mathbb{R}^+$ et $U_{s_i}^- \subset \mathbb{R}^-$ sont les parties strictement positive et strictement négative de U_{s_i} .

$PF_{s_i}^{k_i}$ et $NF_{s_i}^{k_i}$ sont respectivement la partie positive et la partie négative de l'ensemble flou $F_{s_i}^{k_i}$.

$F_{M(PF_{s_i})}^{(k_i)}$ les ensembles flous que peut prendre $\tilde{M}_1(\tilde{s}_1)$ lorsque $\tilde{s}_i = PF_{s_i}^{(k_i)}$.

$F_{M(NF_{s_i})}^{(k_i)}$ les ensembles flous que peut prendre $\tilde{M}_2(\tilde{s}_2)$ lorsque $\tilde{s}_i = NF_{s_i}^{(k_i)}$.

Étape 4. Calcul de la loi de commande floue

Il en résulte donc, d'après les deux dernières équations, la loi de commande floue suivante :

$$\tilde{v}_1 = \left(-\tilde{f}_1(\tilde{z}_1) \tilde{+} \tilde{z}_{ref1} \tilde{+} \frac{1}{\lambda_1} \tilde{\sim} \tilde{s}_{d1}(\tilde{e}_1) \tilde{+} \frac{1}{\lambda_1} \tilde{\sim} \tilde{e}_1 \right) \tilde{+} \tilde{g}_1(\tilde{z}_1) \quad (4.73)$$

$$\tilde{v}_2 = \left(-\tilde{a} \tilde{+} \tilde{z}_{ref2} \tilde{+} \frac{1}{\lambda_2} \tilde{\sim} \tilde{s}_{d2}(\tilde{e}_2) \tilde{+} \frac{1}{\lambda_2} \tilde{\sim} \tilde{e}_2 \right) \tilde{+} \tilde{g}_2(\tilde{z}_1)$$

Cette équation nous permet de déduire les ensembles flous $F_{u_i}^{(k_1, k_2)}$, tels que :

$$F_{v_1}^{(k_1, k_2)} = \left(-F_{f_1}^{(k_1)} \tilde{+} \tilde{z}_{ref1} \tilde{+} \frac{1}{\lambda_1} \tilde{\sim} F_{s_{d1}}^{(k_1)} \tilde{+} \frac{1}{\lambda_1} \tilde{\sim} F_{e_1}^{(k_1)} \right) \tilde{+} F_{g_1}^{(k_1)} \quad (4.74)$$

$$F_{v_2}^{(k_1, k_2)} = \left(-\tilde{a} \tilde{+} \tilde{z}_{ref2} \tilde{+} \frac{1}{\lambda_2} \tilde{\sim} F_{s_{d2}}^{(k_2)} \tilde{+} \frac{1}{\lambda_2} \tilde{\sim} F_{e_2}^{(k_2)} \right) \tilde{+} F_{g_2}^{(k_1)}$$

et la base de règles correspondant à l'entrée de commande u pour chaque sous-système :

$$\begin{aligned} \mathfrak{R}_{v_1}^{(k_1)} : & \quad \text{IF} \quad (z_1, e_1)' \quad \text{is} \quad (F_{z_1}^{k_1}, F_{e_1}^{k_1})' \\ & \quad \text{THEN} \quad v_1 \quad \text{is} \quad F_{v_1}^{(k_1)} \end{aligned} \quad (4.75)$$

$$\begin{aligned} \mathfrak{R}_{v_2}^{(k_1, k_2)} : & \quad \text{IF} \quad (z_1, e_2)' \quad \text{is} \quad (F_{z_1}^{k_1}, F_{e_2}^{k_2})' \\ & \quad \text{THEN} \quad v_2 \quad \text{is} \quad F_{v_2}^{(k_1, k_2)} \end{aligned} \quad (4.76)$$

IV.3.4. Résultats de simulation

Une simulation numérique de la commande à structure variable floue basée sur le modèle flou linguistique du moteur asynchrone est effectuée. Afin de pouvoir comparer les résultats, les mêmes essais que ceux de la commande à structure variable réelle ont été réalisés pour des nombres de partitions différents (3, 5, 7, 9, 11 et 15).

La comparaison entre ces deux commandes est illustrée par les figures 4.13 et 4.14. La plage d'incertitudes n'est pas considérée dans l'élaboration du modèle flou. Ce test confirme que l'approximation est d'autant plus précise que le nombre de partitions de l'espace d'entrée augmente (l'approximation est meilleure à partir d'un partitionnement de 7 ensembles flous).

Pour montrer l'effet de cette précision sur le comportement du système, nous avons effectué une simulation de la commande relative à chaque partition, tout en gardant les mêmes conditions d'essai. Les résultats obtenus sont présentés dans la figure 4.15. Les allures des réponses des différentes grandeurs ressemblent à celles obtenues dans la section §II.4. Nous remarquons, dans ce cas aussi, que plus le nombre de partitions augmente, plus la commande robuste floue est susceptible d'approximer la commande à structure variable conventionnelle.

De la même façon que pour le cas réel, un test de robustesse a été effectué pour un nombre de partitions égal à 3, 5 et 7, pour une variation de la résistance rotorique de 50% jusqu'à 200% par rapport à la résistance nominale, et pour un couple de charge non estimé. La plage des incertitudes est prise en compte par le modèle flou. Les réponses obtenues sont représentées par les figures 4.16-4.17. Les résultats obtenus sont similaires à ceux de la figure 2.18, en indiquant toujours une bonne robustesse. Ces résultats montrent également que le contrôleur utilisant le modèle flou avec un partitionnement de 3 ensembles flous de l'univers de discours des variables d'entrée est robuste, malgré la mauvaise approximation du modèle original. Cela prouve que le modèle proposé dans sa forme canonique est efficace, et que la commande à structure variable floue a bien pris en compte la plage des incertitudes.

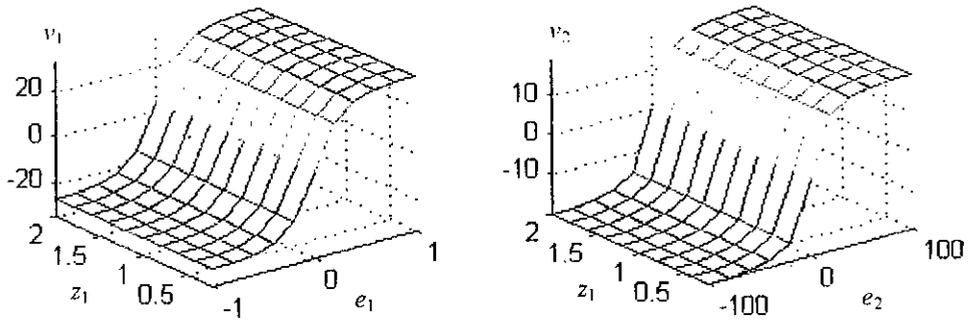
IV.4. Conclusion

Dans ce chapitre, nous avons présenté deux façons d'introduire la modélisation floue dans la commande d'un moteur asynchrone. La modélisation floue adoptée est de type linguistique, elle exploite la connaissance préalable du modèle analytique décrivant grossièrement le comportement du système incertain.

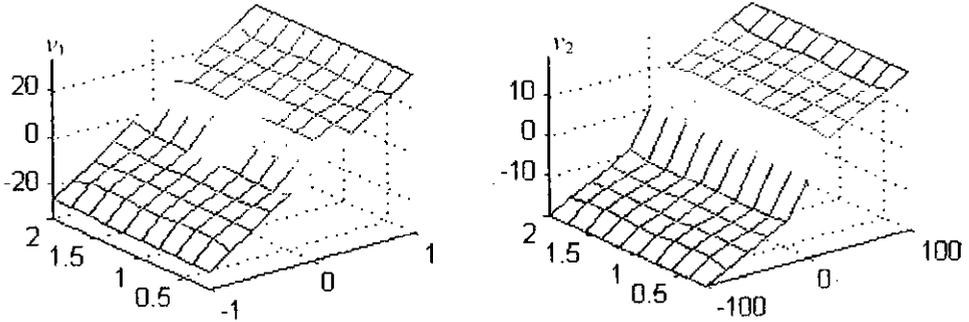
Dans un premier cas, un modèle inverse flou a été utilisé pour remplacer le modèle inverse dans la commande à flux orienté classique. L'utilisation d'un modèle linguistique offre plus de transparence dans le réglage qui se fait après la mise en œuvre de la stratégie de commande.

Le second cas concerne l'obtention du modèle flou linguistique du moteur asynchrone présenté dans le référentiel statorique sous forme canonique. Ce modèle prend en charge toutes les incertitudes (paramétriques, non-paramétriques et structurelles), afin d'assurer une bonne robustesse lors de son introduction dans la conception de la commande floue robuste.

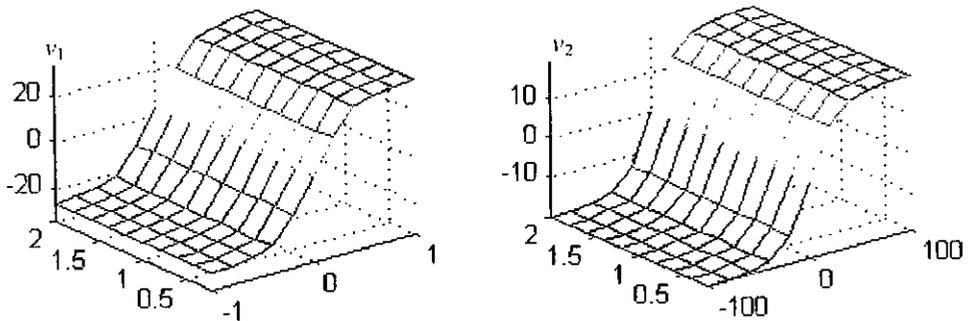
La commande à structure variable floue basée sur le modèle flou du système a été présentée pour une large gamme de modèles, puis pour les systèmes mis sous forme canonique, avant de l'appliquer dans le cas du moteur asynchrone pour le réglage du flux et de la vitesse rotoriques.



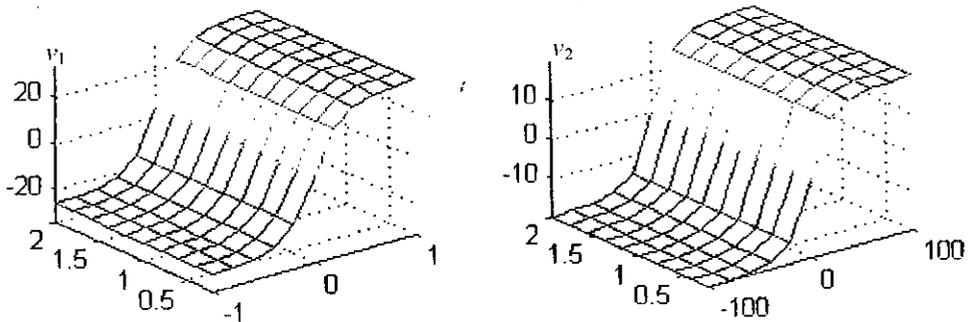
a) sorties de commande pour le cas réel



b) sortie de commande floue pour $M_i = 3$



c) sortie de commande floue pour $M_i = 7$



d) sortie de commande floue pour $M_i = 15$

Figure 4.13 : Commande à structure variable floue. Comparaison entre les commandes réelle et floue pour différentes valeurs de M_i

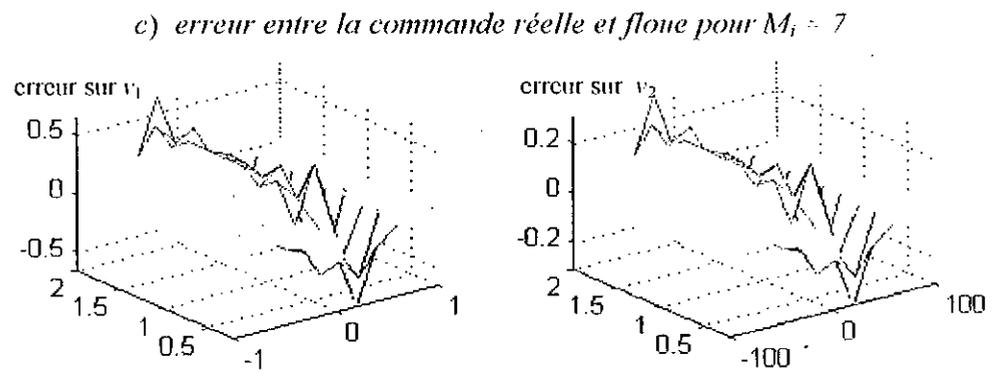
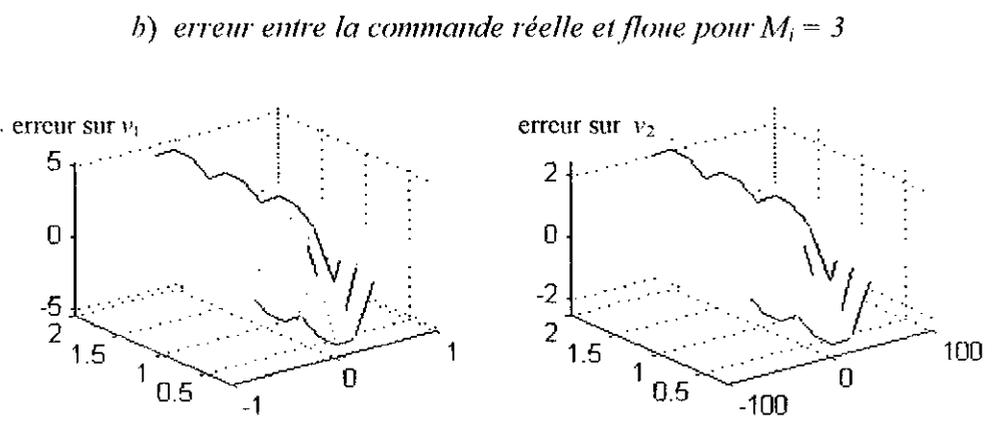
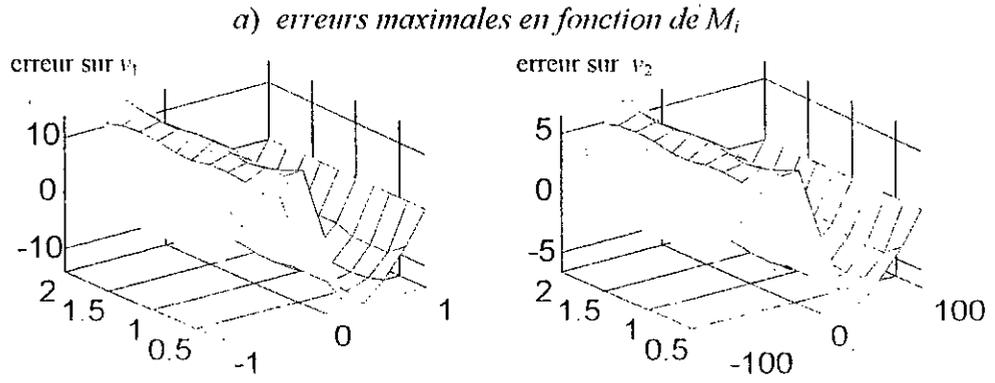
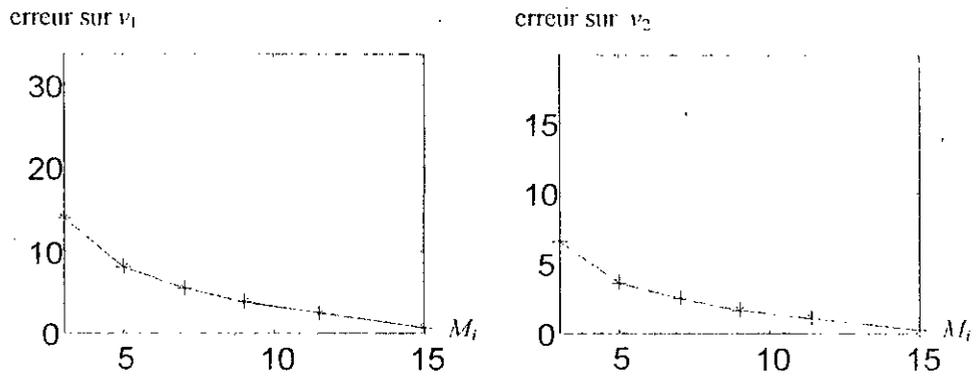


Figure 4.14 : Commande à structure variable floue. Comparaison entre les commandes floue et réelle pour différentes valeurs de M_i

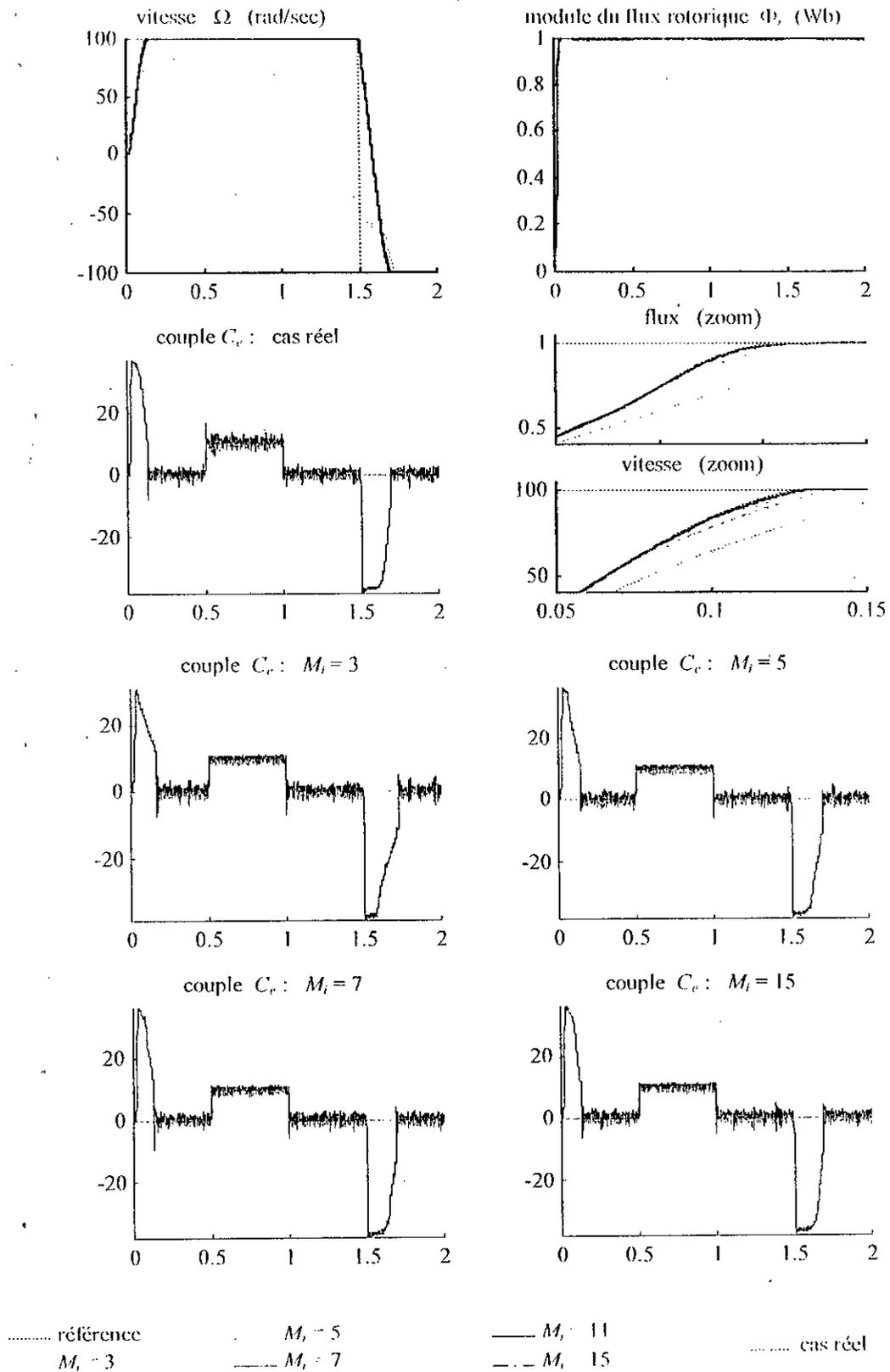
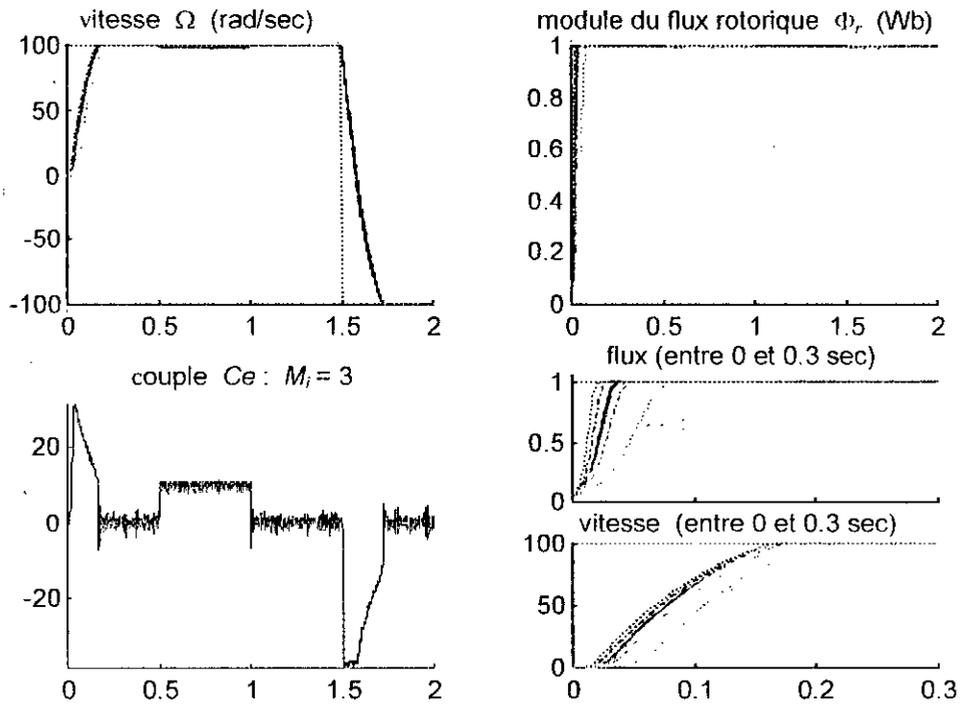
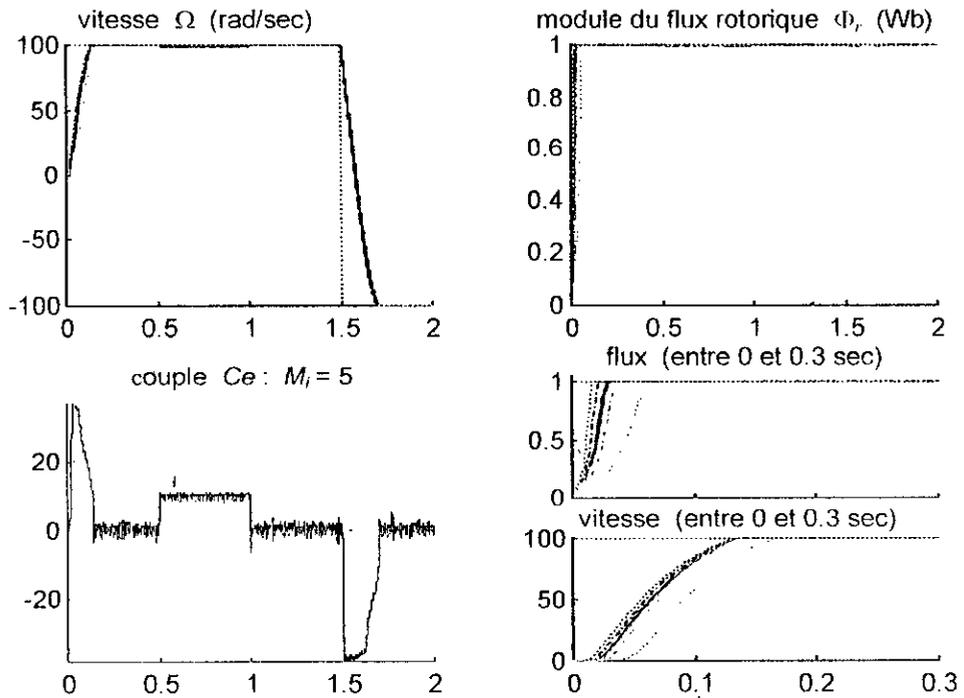


Figure 4.15 : Commande à structure variable floue. Résultats de simulation pour différentes valeurs de M_i (avec estimation de la charge et sans variation des paramètres)



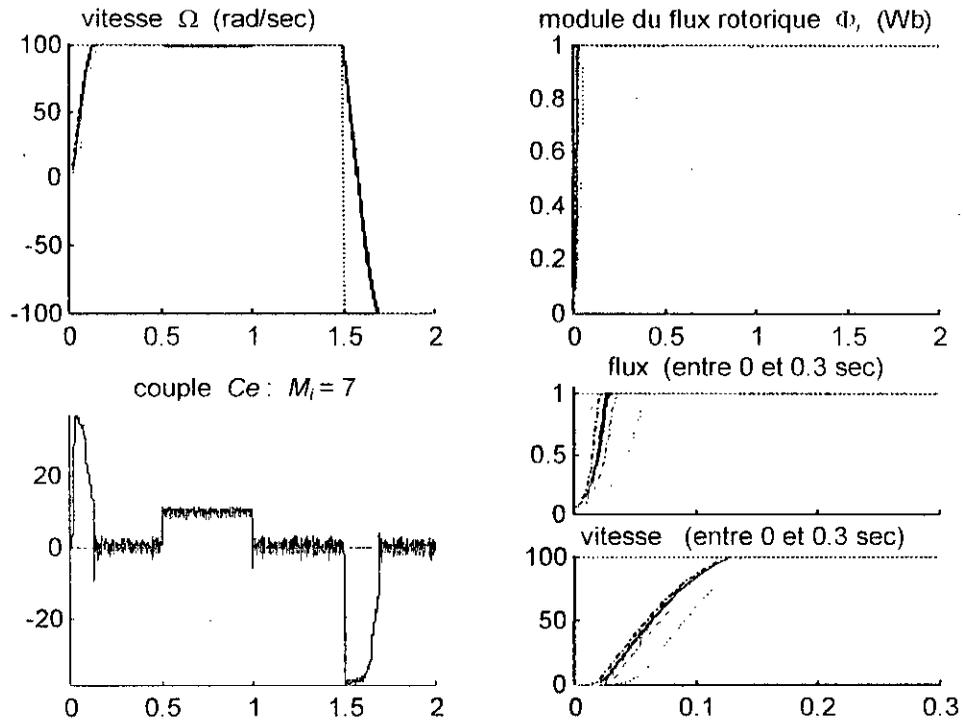
a) avec un partitionnement de l'espace d'entrée de 3 ensembles flous



b) avec un partitionnement de l'espace d'entrée de 5 ensembles flous

..... référence $R_r = 90\% R_{r0}$ — $R_r = 110\% R_{r0}$ $R_r = 200\% R_{r0}$
 $R_r = 50\% R_{r0}$ — cas idéal - - - $R_r = 150\% R_{r0}$

Figure 4.16 : Commande à structure variable floue. Résultats de simulation pour différentes variations de R_r (sans estimation de charge et avec une variation de 5% sur les inductances)



..... référence
 $R_r = 90\% R_{r0}$
 $R_r = 110\% R_{r0}$
 $R_r = 200\% R_{r0}$
 $R_r = 50\% R_{r0}$
 cas idéal
 $R_r = 150\% R_{r0}$

Figure 4.17 : Commande à structure variable floue. Résultats de simulation pour différentes variations sur R_r (avec un partitionnement de l'espace d'entrée de 7 ensembles flous, sans estimation de charge et avec une variation de 5% sur les inductances).

Conclusion générale

La commande du moteur asynchrone reste jusqu'à ce jour un problème contrariant pour les chercheurs. La difficulté réside essentiellement dans son modèle mathématique caractérisé par :

- ▀ une forte non-linéarité et un couplage étroit entre les variables internes (couple et flux).
- ▀ la présence des incertitudes et la variation des paramètres dues à l'effet thermique et au vieillissement de la machine.
- ▀ la difficile accessibilité aux variables d'état (flux rotorique).

Par rapport à la complexité et à l'incertitude du système, des travaux ont été consacrés pour les méthodes de simplification et d'imitation des parties difficilement réalisables du modèle et/ou de la commande. Entre autres, on peut citer la logique floue, dont l'intérêt réside dans sa capacité de résoudre des problèmes complexes par des solutions simples.

L'objectif de notre travail est de trouver un meilleur moyen pour introduire la logique floue dans la stratégie de commande robuste d'un moteur asynchrone, tout en gardant les mêmes, voire de meilleures performances que celles des techniques conventionnelles.

Pour ce faire, quelques stratégies classiques de commande du moteur asynchrone ont été d'abord étudiées, à savoir la commande par flux orienté, la commande par placement de pôles basée sur un retour d'état linéarisant le modèle, et la commande à structures variables.

La commande à flux orienté a montré son efficacité dans le découplage, dans le régime permanent, entre les variables (couple et flux), mais elle présente cependant l'inconvénient d'être très sensible aux variations des paramètres, surtout celle de la constante de temps rotorique.

Le retour d'état linéarisant le modèle de la machine permet d'une part, l'application des techniques de réglages pour les systèmes linéaires (telle que la commande par placement de pôles), et d'autre part une amélioration des performances au niveau du découplage qui est

réalisé quel que soit le régime de fonctionnement. En plus, du point de vue implémentation, la commande se fait dans le référentiel statorique qui est plus simple à réaliser. Néanmoins, son inconvénient majeur réside dans sa sensibilité aux incertitudes inhérentes au système, surtout lorsqu'il s'agit des perturbations externes.

La commande à structure variable apporte de meilleures performances, principalement en ce qui concerne la robustesse, par rapport aux deux commandes précédentes. Le temps de réponse ainsi que le temps de calcul sont également améliorés. Ces résultats sont prévisibles de par la nature de cette commande. Un autre avantage appréciable est qu'elle ne nécessite, au préalable, ni l'estimation de la résistance rotorique, ni celle du couple résistant. Par contre, une connaissance précise du flux est exigée, d'où la contrainte d'utilisation d'un bon observateur du flux.

L'inconvénient majeur de cette technique de commande est le phénomène de chattering. La réduction de ce dernier est généralement effectuée par le choix d'une relation linéaire, par partie, du gain de la commande discontinue. De meilleures performances peuvent être obtenues dans le cas d'une non-linéarité de cette fonction, mais au prix d'une difficulté dans sa conception et dans sa réalisation. En outre, elle est gourmande vis-à-vis du temps de calcul. Ce qui nous a incité à introduire la logique floue dans cette stratégie de commande.

Les notions de base de la logique floue, des systèmes flous et des contrôleurs flous ont été également présentées. Nous avons ensuite procédé à une étude synthétique ayant pour but d'aider le concepteur à prendre un choix convenable lors de l'élaboration du contrôleur flou approprié à son système. Ce choix se fait selon la prise ou non en considération de l'expression analytique du modèle, des données expérimentales, de son expérience, ou à partir du modèle flou du système que nous avons retenu.

Par la suite, nous avons étudié la modélisation floue d'une manière générale. Une approche de modélisation floue pour les systèmes incertains a été développée et présentée. Cette approche permet de convertir une représentation analytique du système incertain en une représentation basée sur des règles linguistiques de type "If-Then". Ce type de modélisation offre l'avantage d'être adapté à la conception des systèmes de commande robustes du fait qu'il tient compte des plages de variation des paramètres.

Afin de bénéficier des avantages que procure la modélisation floue, nous avons présenté une approche de commande par flux orienté en se basant sur un modèle inverse flou du moteur. Nous avons, dans ce cas, converti la représentation analytique du bloc de découplage dans la commande à flux orienté en une représentation linguistique floue. Le modèle résultant garde les mêmes inconvénients que celui de la CFO, à savoir la sensibilité aux incertitudes et aux variations paramétriques liées au moteur asynchrone. Néanmoins, il présente une meilleure simplicité et fiabilité de réglage après la mise en œuvre de la commande.

Nous avons de même appliqué une commande à structure variable floue au modèle flou du moteur. Pour cela, un modèle mis sous une forme canonique basé sur un retour d'état a été proposé. Cette représentation consiste à simplifier au maximum la boucle interne, qui est l'origine du découplage entre les variables à contrôler, tout en gardant une forme canonique du système, permettant ainsi une commande décentralisée. Le système résultant a été, par la suite, représenté par un modèle flou susceptible de fournir une commande robuste floue. L'idée de cette approche consiste en une extension des techniques de la commande à structure variable classique pour les systèmes dynamiques flous. La commande ainsi obtenue n'est pas constituée d'une combinaison entre des expressions réelles et floues, mais elle est constituée intégralement à base de règles floues. La robustesse pour ce genre de commandes est toujours assurée.

L'avantage principal d'une commande robuste floue basée sur un modèle flou linguistique est qu'elle prend en charge toutes les incertitudes paramétriques et non paramétriques, sans avoir à considérer leur forme, leur structure ou leur répartition spectrale ; ce qui n'est pas le cas dans plusieurs stratégies de commande des systèmes probabilistes. D'autre part, ceci nous a permis d'éviter le calcul préalable des paramètres de la commande (par exemple le gain de la commande discontinue dans le cas de la CSV) en fonction de ces incertitudes.

L'approche de la modélisation floue offre aussi l'avantage, d'une part, de conserver les performances de la commande indépendamment de la complexité du modèle en faisant seulement un bon choix de partitionnement et des formes des ensembles flous des entrées / sorties, et d'autre part, d'assurer une simplicité d'implémentation, de rapidité de calcul et de réduire la capacité mémoire.

Suggestions et perspectives

La méthode proposée peut être appliquée sur une large classe de systèmes incertains. Son choix repose sur la présence d'un modèle analytique, qui peut être non-linéaire et de représentation complexe.

Le nombre d'entrées de commande, par contre, doit être le moins élevé possible. Dans le cas contraire, il est fortement recommandé de décomposer le modèle d'origine en sous-systèmes, et de réduire la base de règles du modèle flou approprié [Set95][Set98b].

Dans le cas d'un système couplé, la décomposition par retour d'état permettant de le mettre sous une forme canonique de la même manière que celle présentée auparavant, est suggérée.

Une fois le modèle (ou le contrôleur) flou réalisé, un réglage postérieur peut être effectué par des algorithmes d'optimisation dédiés pour les systèmes flous [Isa92], tels que la programmation évolutionnaire (en particulier les algorithmes génétiques [Kim95]) et les

algorithmes adoptés dans les systèmes neuronaux (qui nécessitent de convertir le système flou en une architecture neuro-floue [Jan92]).

Concernant l'estimation des composantes du flux, des travaux de recherche ont été consacrés aux observateurs non-linéaires du moteur asynchrone. On cite [Tad99a] et [Tad99b] où la technique des modes de glissement permet une bonne observation des composantes du flux dans le référentiel fixe, sans avoir besoin d'une précision sur les paramètres du moteur.

Bibliographie

- [Bab98a] P. Baranyi, I.M. Bavelaar, R. Babuska, *et al.* "A method to invert a linguistic fuzzy model", *Int. J. of Systems Science*, vol. 29, n° 7, pp.711-721, 1998.
- [Bab98b] R. Babuska, "Fuzzy modeling: Principles, methods and applications.", *In* C. Bonivento, C. Fantuzzi, R. Rovatti, "*Fuzzy Logic Control: Advances in Methodology*", *World Scientific*, Singapore, pp.187-220, 1998.
- [Bag96] L. Baghli, H. Razik, A. Rezzoug, "Régulation floue de la vitesse dans une commande vectorielle de machine asynchrone par orientation du flux rotorique", *LFA'96*, Nancy, France, pp.40-46, 4-5 déc. 1996.
- [Bar82] P. Barret, "Regimes transitoires des machines tournantes électriques", *Ed. Eyrolles*, 1982.
- [Bar93] L. Barazane, "Commande vectorielle d'une machine asynchrone alimenté en courant", *Thèse de magister, ENP, Alger*, 1993.
- [Bar98] G. Bartolini, M. Marchesoni, P. Pisu, E. Usai, "Chattering reduction and robust position control in induction motor with second-order VSS". *Int. Journal of Systems Science*, vol.29, n°1, pp.1-12, 1998.
- [Bar00a] L. Barazane, C. Larbes, R. Ouiguini, Y. Sellami, "Commande hybride, linéarisante et à structures variables d'un moteur asynchrone", *ICEE'2000, 1st Inter. Conf. on Electrical Engin.*, Boumerdes, Algérie, 4-6 Nov. 2000, scope C11.
- [Bar00b] L. Barazane, R. Ouiguini, C. Larbes, Y. Sellami, "Commande hybride, linéarisante et à structures variables d'un moteur asynchrone alimenté par un onduleur MLI à hystérésis des courants redressés", *ICEL'2000, UST. Oran, Algérie*, vol.1, 13-15 nov. 2000, pp.190-195.
- [Bar00c] L. Barazane, "Contribution à la commande en cascade de la machine asynchrone par l'application des techniques de l'intelligence artificielle", *ENP, Alger, Algérie*, (à apparaître)
- [Bla71] F. Blaschke, "A new method for the structure decoupling of AC induction machines", *Proc. 2nd IFAC Symp. Multivariable Techn. Control Systems, Part 3*, pp.11-13, oct.1971.

- [Bla72] F. Blaschke, "The principle of field orientation as applied to the new transvector", *Siemens review*, vol. 34, pp.217-222, may 1972.
- [Bod94] M. Bodson, J. Chiasson, and R. Novotnak, "High performance of induction motor control via input-output linearization", *IEEE Control System*, pp.25-33, August 1994.
- [Bor96] P. Bortolet, S. Boverie, A. Titli, "Modélisation floue et contrôle de systèmes non linéaires", *LFA'96-Nancy*, France, pp.106-133, dec. 1996.
- [Bou95] B. Bouchon-Meunier, "La logique floue et ses applications", *Ed. Addison-Wesley*, France, 1995.
- [Bou99a] M. Boukhniher, "Commande robuste H_∞ d'une machine asynchrone alimentée par un onduleur multiniveaux à structure NPC", *Thèse de Magister, ENP*, Alger.
- [Bou99b] M.E.K. Bouyoucef, "La commande à structures variables par régime glissant et la logique floue appliquée à un robot manipulateur", *Thèse de Magister, USTHB*, Alger, 1999.
- [Bra96a] P.J.Costa Branco, N. Lori, J.A. Dente, "New approaches on structure identification of fuzzy models: case study in an electro-mechanical system", in "Fuzzy Logic, Neural Networks, and Evolutionary Computation", Eds. T. Furuhashi and Y. Uchikawa, pp. 104-143, *Springer-Verlag*, Berlin, 1996.
- [Bra96b] P.J.Costa Branco, J.A. Dente, "Inverse-model compensation using fuzzy modeling and fuzzy learning schemes". In "Intelligent engineering systems through artificial neural networks, Smart engineering systems: Neural Networks, Fuzzy Logic and Evolutionary Programming", Ed. C.H. Dagli, M. Akay, C.L. Philip Chen, B. Fernández, and J. Ghosh, vol.6, ASME Press, New York, pp. 237-242, 1996.
- [Car94] A. De Carli, P. Liguori, A. Marroni, "A fuzzy-PI control strategy", *Control Eng. Practice*, vol.2, n°1, pp.147-153, 1994.
- [Cha81] L. Chambadal, "Dictionnaire de mathématiques", *Hachette*, Paris, France, 1981.
- [Cha84] R. Chauprade, F. Milsant, "Electronique de puissance 2. Commande des moteurs à courant alternatif", *Eyrolles*, Paris, 1981.
- [Cha90] J. Chatelain, "Machines électriques", *Presses Polytechniques normandes*, 1990.
- [Cha93] T.Y. Chang, C.M. Hong, C.T. Pan, "A no chattering discrete-time sliding mode controller for field acceleration method induction motor drives", *24th Annual IEEE Power Elect. Specialists Conf., PESC'93*, pp.1158-1164, 1993.
- [Che00] H. Chekireb, M. Tadjine, "Commande flou adaptative basée sur l'identification inverse: Application au moteur asynchrone", *ICEE'2000, 1st Inter. Conf. on Electrical Engin.*, Boumerdes, Algérie, 4-6 Nov. 2000, scope B09.
- [Che00] H. Chekireb, M. Tadjine, "Commande non-linéaire par la méthode de Lyapounov de la machine asynchrone", *ICEL'2000, U.S.T. Oran*, Algérie, vol.1, 13-15 nov. 2000, pp.116-120.

- [Chu95a] S.K. Chung, J.H. Lee, J.S. Ko, M.J. Youn, "Current control of voltage-fed PWM inverter for AC machine integral variable structure control". *Proc. of the 1995 IEEE IECON 21st Inter. Conf. on Ind. Electronics*, vol.1, pp.668-673, 1995.
- [Chu95b] S.K. Chung, J.H. Lee, J.S. Ko, M.J. Youn, "Robust speed control of brushless direct-drive motor using integral variable structure control", *IEE Proc. Electric Power Applications*, vol.142-6, pp.361-370, Nov. 1995.
- [Dec88] R.A. De Carlo, S.H. Zak, G.P. Matthews, "Variable structure control of nonlinear multivariable systems: a tutorial", *Proc. IEEE*, vol.76, n°3, pp.212-232, Mar. 1988.
- [Dri93] D. Driankov, H. Hellendoorn, M. Reinfrank, "An introduction to fuzzy control". *Springer-Verlag*, Berlin, 1993.
- [Dri96] R. Palm, D. Driankov, H. Hellendoorn, "Model-based fuzzy control: Fuzzy gain schedulers and sliding mode fuzzy controllers", *Springer Verlag*, Berlin, 1996.
- [Eco86] C.G. Economou, M. Morari, B.O. Palsson, "Internal model control", *Ind. & Engin. Chemistry, Process Design and Development*, vol.25, pp.403-411, 1986.
- [Eme62] Taran Emelyanov, "Sur une classe de systèmes de régulation automatique à structure variable", *Journal de l'académie des sciences d'URSS, Énergétique et automatique*, n°3, 1962.
- [Fil60] A. Filippov, "Equations différentielles à second membre discontinu", *Journal de mathématiques*, Tome 51, n°1, 1960.
- [Fis98] M. Fischer, O. Nelles, R. Isermann, "Predictive control based on local linear fuzzy models", *Int. J. of Systems Science*, vol.29, n°7, pp.679-697, 1998.
- [Fu91] Y. Fu, "Commandes découplées et adaptatives de machines asynchrones 3~", *Thèse de Doctorat, Univ. Montpellier II, France*, mars 1991.
- [Gha94] M. Ben-Ghali, A.T. Alouani, "Fuzzy modelling of dynamic systems", *Proc. of the 26th Southern Symposium on System Theory*, pp.168-171, 1994.
- [Gha95b] M. Ben-Ghali, A.T. Alouani, "A fuzzy variable structure approach to feedback regulation of uncertain dynamical systems, with application to robotics", *Information Sciences, Elsevier Science*, vol.85, pp.241-273, 1995.
- [Gha96a] M. Ben-Ghali, "Nonlinear modeling of uncertain dynamical systems using fuzzy set concepts", *Proc. of the 35th IEEE Conf. on Decision and Control*, Kobe, Japan, pp.418-423, December 11-13, 1996.
- [Gha96b] M. Ben-Ghali, "Robust model-based control of uncertain dynamical systems : a fuzzy set theory based approach", *Proc. of the 35th Conf. on Decision and Control*, vol.1, Kobe, Japan, pp.807-812, Dec.1996.
- [Gha96c] M. Ben-Ghali, "Aspects of the effects of the defuzzification stage on the stability and performance of fuzzy model-based control systems", *Proc. of the 5th Int. Conf. on Fuzzy Systems*, vol.1, pp.475-478, 1996.

- [Gha97a] M. Ben-Ghalia, "Modelling and robust control of uncertain dynamical systems using fuzzy set theory", *Int. J. Control*, vol.68, n°6, pp. 1367-1395, 1997.
- [Gha97b] W. Gharieb, M.A. Shejrah, "Fuzzy retuning PI controller", *IFAC-IFIP-IMACS Conference on control of industrial systems*, Belfort, France, pp.167-171, 20-22 May 1997.
- [Hal69] J.L. Hale, "Ordinary differential equations", Wiley-Interscience, New York, 1969.
- [Ham93] M. Hamerlain, "Commande hiérarchisée à modèle de référence et à structures variables d'un robot manipulateur à muscles artificiels", *Thèse n° 223, GARI / DGE / INSA*, Toulouse, Janvier 1993.
- [Har86] F. Harashima, H. Hashimoto, K. Maruyama, "Practical robust control of robot arm using variable structure systems", *Proc. of IEEE, Int. Conf. on robotics and automation*, San Francisco, pp.532-538, 1986.
- [He97] G.D. He, J.P. Jiang, J.X. Shen, J. Lei, "A fuzzy variable structure control scheme for AC servo drive system", *IEEE International Electric Machines and Drives Conference (IEMDC'97)*, Milwaukee, USA, pp.WC3/3.1-WC3/3.3, May 1997.
- [Hir95] K. Hirata, "History of industrial applications of fuzzy logic in Japan", chapter 2, in J.Yen, W.R.Langari, L.A.Zadeh, "*Industrial Applications of Fuzzy Logic and Intelligent Systems*", IEE Press, pp.43-54, 1995.
- [Hsu98] Y.C. Hsu, H.A. Malki, "Fuzzy variable structure control for MIMO systems", *IEEE Int. Conf. Fuzzy Syst. Proc.*, vol.1, pp.280-285, 1998.
- [Igo95] S. Igor, Z. Milan, "Direct field-oriented induction motor drive with nonlinear flux observer", *IFAC Motion Control*, Germany, pp.493-500, 1995.
- [Isa92] S. Isaka, A.V. Sebald, "An optimization approach for fuzzy controller design", *IEEE Trans. Syst. Man & Cyb.*, vol.22, n°6, pp.1469-1473, Nov./Dec.1992.
- [Ish93] A. Ishigame, T. Furukawa, S. Kawamoto, T. Taniguchi, "Sliding mode controller design based on fuzzy inference for nonlinear systems", *IEEE. Trans. Indust. Electronics*, vol.40, n°1, pp.64-70, 1993.
- [Isi89] A. Isidori, "Nonlinear Control Systems - An Introduction", *Springer Verlag*, Berlin, 1989.
- [Jan92] J.S.R. Jang, "Neuro-Fuzzy Modeling: Architectures, Analyses, and Applications", *Ph.D. Dissertation, EECS Department, Univ. of California at Berkeley*, 1992.
- [Jan95] J.S.R. Jang, C-T. Sun, "Neuro-Fuzzy Modeling and Control", *The Proc. of the IEEE*, vol. 83, pp. 378-406, Mar. 1995.
- [Kan98] H.J. Kang, C. Kwon, H. Lee, M. Park, "Robust stability analysis and design method for the fuzzy feedback linearization regulator", *IEEE Trans. Fuzzy Systems*, vol.6, n°4, pp.464-472, Nov.1998.
- [Kaz96] M.P. Kazmierkowski, D.L. Sobczuk, "Sliding mode feedback linearized control of PWM inverter fed induction motor", *IEEE IECON'96*, vol.1, pp.244-249, 1996.

- [Kel97] W.E. Kelly, "Dimensionality in fuzzy systems", *Dissertation, Texas A&M university, Texas*, Aug. 1997.
- [Kim95] J. Kim, Y. Moon, B.P. Zeigler, "Designing fuzzy net controllers using genetic algorithms", *IEEE Control Systems*, pp.66-72, 1995.
- [Kim97] E. Kim, M. Park, S. Ji, M. Park, "A new approach to fuzzy modelling", *IEEE Tans. on Fuzzy Systems*, vol.5, n°3, pp.328-337, 1997.
- [Kóc97] L.T. Kóczy, K. Hirota, "Size reduction by interpolation in fuzzy rule bases", *IEEE Trans. Syst. Man. & Cybernetics, Part-B*, vol.27, n°1, pp. 14-25, Feb. 1997.
- [Kre96] V. Krebs, R. Böhm, "Logical design of relational fuzzy controllers", *2nd World Automation Congress, WAC'96, Montpellier 1996*.
- [Kun94] Ying-Shieh Kung, Chang-Ming Liaw, "A Fuzzy Controller Improving a Linear Model Following Controller for Motor Drives", *IEEE Transactions on Fuzzy Systems*, vol. 2, n° 3, pp.194-202, Aug. 1994.
- [Lal99] D. Lalili, "Sur les commandes non-linéaires adaptatives de la machine asynchrone". *Thèse de Magister, ENP, Alger, 1999*.
- [Lan95] N. Langovsky, M. Elbuluk, D. Kankam, "Nonlinear flux observer with on-line parameter tuning for wide speed operation of induction machines", *Proceedings of The IEEE Industry Application Society 30th Annual Meeting (IAS'95)*, Orlando, Florida, pp.144-151, October 1995
- [Lec90] C.C. Lee, "Fuzzy logic in control systems: fuzzy logic controller-part 1 (and part 2)", *IEEE Trans. on Systems, Man, and Cybernetics*, vol.20, n°2, pp.404-418 (and pp.419-435), 1990.
- [Li97] H.X. Li, H.B. Gatland, A.W. Green, "Fuzzy variable structure control", *IEEE Trans. Syst. Man. Cyb.*, vol.27, n°2, pp.1085-1087, April 1997.
- [Lu98] Y. Lu, T.Q. Chen, "Fuzzy diagnostic model and its application in automotive engineering diagnosis", *Applied Intelligence, Kluwer Academic Publishers*, vol.9, pp.231-243, 1998.
- [Lyg94] J. Lygeros, D.N. Godbole, C.P. Coleman, "Model based fuzzy logic control", *33rd IEEE CDC*, pp.2633-2638, 1994. (Technical Note, Electronics Research Laboratory, U.C.Berkeley, UCB/ERL-M90/60).
- [Mad96] N. Madani, "Contribution à l'étude de la commande vectorielle avec réglage linéaire et par mode de glissement de la machine asynchrone", *Thèse de Magister, Univ. Tizi-Ouzou, Tizi-Ouzou, 1996*.
- [Mam75] E.H. Mamdani, S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller", *Int. J. of Man-Machine Studies*, vol.7, n°1, pp.1-13, 1975.
- [Män99] M. Männle, "Identifying Rule-Base TSK fuzzy models", *Proc. EUFIT, Aachen, Germany*, pp.286-299, Sep.1999.
- [Mar91] R. Marino and P. Valigi, "Nonlinear control of induction motors : a simulation

- study", *In European Control Conference*, Grenoble, France, pp.1057-1062, 1991.
- [Mar97] M. Marchesoni, P. Segarich, E. Soressi, "A simple approach to flux and speed observation in induction motor drives", *IEEE Trans. Ind. Electronics*, vol.44, n°4, pp.528-535, Aug.1997.
- [Mat99] "Fuzzy logic Toolbox for use with Matlab. User's guide", Version 2, *The Math Works*, 1999.
- [Men95] J.M. Mendel, "Fuzzy logic systems for engineering : a tutorial", *Proc. of the IEEE*, vol.83, n°3, pp.345-377, march 1995.
- [Mid00] A. Midoun-Oussedik, "La commande robuste à structure variable et à logique floue appliquée à un moteur asynchrone", *Magister, EMP*, Algérie, 2000.
- [Mor85] G.R. Morgan, Ü. Özgüner, "A decentralized variable structure control algorithm for robot manipulators", *IEEE, J.Robotics Automation*, vol.1, n°1, pp.57-65, 1985.
- [Nib94] M. Nibouche, "Application de commandes nonlinéaires au moteur asynchrone", *Thèse de Magister, ENP*, Alger, Juillet 1994.
- [Pan94] C.T. Pan, T.Y. Chang, "A fixed structure sliding mode controlled induction motor drive", *IEEE Trans. Energy Conversion*, vol.94, pp.243-249, 1994.
- [Ped84] W. Pedrycz, "An identification of systems in fuzzy relational systems", *Fuzzy Sets and Systems*, vol.13, pp.153-167, 1984.
- [Ped97] W. Pedrycz, M. Reformat, "Rule-Based Modeling of Nonlinear Relationships", *IEEE Trans. Fuzzy Syst.*, vol.5, n°2, May 1997.
- [Rud64] Walter Rudin, "Principles of mathematical analysis", *McGraw-Hill*, New York, 1964.
- [Sab81] A. Sabanovic, D.B. Izosimov, "Application of sliding mode to induction motor control", *IEEE Trans Ind. Applicat*, vol IA 17, n°1, pp.41-49, Jan / Feb 1981.
- [Sab89] A. Sabanovic, F. Bilalovic, "Sliding mode control of AC drives", *IEEE Trans Ind. Applicat.*, vol 25, n°1, pp.70-75, Jan / Feb 1989.
- [Sel99] Y. Sellami, R. Ouiguini, C. Larbes, "Nouvelle approche de la commande par flux orienté d'un moteur asynchrone, utilisant un modèle inverse flou", *CSCA '99*, Alger, nov. 1999.
- [Set95] M. Setnes, "Fuzzy rule-base simplification using similarity measures", *M.Sc. Thesis, Delft University of Technology*, Delft, Netherlands, July 1995.
- [Set98a] M. Setnes, R. Babusuka, U. Kaymak, H.R. Nauta Lemke, "Similarity measures in fuzzy rule base simplification", *IEEE Trans. Syst. Man. & Cyb., Part-B*, vol.28, n°3, pp.376-386, June 1998.
- [Set98b] M. Setnes, R. Babuska, H.B. Verbruggen, "Complexity reduction in fuzzy modeling", *Mathematics & Computers in Simulation*, vol.46, pp.507-516, 1998.
- [Sin99] B.N. Singh, B. Singh, B.P. Singh, "Fuzzy control of integrated current-controlled

- converter-inverter-fed cage induction motor drive", *IEEE Trans. Indust. Applic.*, vol.35, n°2, pp.405-412, March/April 1999.
- [Slo84] J.J. Slotine, "Sliding controller design for nonlinear systems", *IGC*, vol.40, n°2, pp.421-434, 1984.
- [Slo91] J.J. Slotine, W. Li, "Applied nonlinear control", *Englewood Cliffs, NJ : Prentice Hall*, 1991.
- [Sob96a] M.P. Kazmierkowski, D.L. Sobczuk, "Sliding mode feedback linearized control of PWM inverter fed induction motor". *IEEE IECON, 22nd Int. Conf. on Ind. Electronics, Control and Instrumentation*, vol.1, pp.244-249, 1996.
- [Sug86] M. Sugeno and G.T. Kang, "Fuzzy modeling and control of multilayer incinerator", *Fuzzy Sets and Systems*, vol.18, pp. 329-346, 1986.
- [Sug88] M. Sugeno, G.T. Kang, "Structure identification of fuzzy model", *Fuzzy Sets and Systems*, North Holland, vol.28, pp.15-33, 1988.
- [Sug92] K. Tanaka and M. Sugeno, "Stability analysis and design of fuzzy control systems", *Fuzzy Sets and Systems*, vol.45, pp. 135-156, 1992.
- [Sug93] M. Sugeno, T. Yasukawa, "A fuzzy logic based approach to qualitative modeling", *IEEE Trans. Fuzzy Syst.*, vol.1, pp.7-31, Feb.1993.
- [Tad99a] A. Bencaib, A. Rachid, E. Audrezet, M. Tadjine, "Real-time sliding-mode observer and control of an induction motor", *IEEE Trans. Ind. Elec.*, vol.46, n°1, pp.128-138, Feb. 1999.
- [Tad99b] A. Bencaib, M. Tadjine, A. Rachid, "Sliding mode control of an induction motor with unknown load: application on a DSP-based system", *Int. Journal of Systems Science*, vol.30, n°8, pp.849-863, 1999.
- [Tak85] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control", *IEEE Trans. Syst., Man, and Cybernetics*, vol.15, pp. 116-132, 1985.
- [Tan94] Y. Tang, L. Xu, "Fuzzy logic application for intelligent control of a variable speed drive", *IEEE Trans Energy Conversion*, vol.9, n°4, pp.679-685, Dec. 1994.
- [Tle99] A. Tlemçani, "Sur la commande adaptative par les systèmes flous: application à la machine synchrone à aimants permanents", *Thèse de Magister, ENP, Alger*, Déc.1999.
- [Ton78] R.M. Tong, "Synthesis of fuzzy models for industrial processes - some recent results", *International Journal of General Systems*, vol.4, pp. 143-163, 1978.
- [Trz94] A.M. Trzynadlowsky, "The field orientation principle in control of induction motor", *Kluwer Academic Publishers*, 1994.
- [Tsu79] Y. Tsukamoto., "An approach to fuzzy reasoning method." In M.M. Gupta, R.K. Ragade, R.R. Yager, éditeurs, "Advances in Fuzzy Set Theory and Applications", North-Holland, Amsterdam, pp.137-149, 1979.

- [Tut95] C. Tuttas, "Voltage-source inverter fed asynchronous machine with hysteresis control of rectified motor currents", *University of Kaiserslautern FB Elektrotechnik*, Kaisers-lautern, Germany, 1995.
- [Uni98] "Encyclopédie universalis", 1998.
- [Utk92] V.I. Utkin, "Sliding mode control and optimization", *Springer-Verlag*, Berlin, 1992.
- [Utk93] V.I. Utkin, "Sliding mode control design principles and applications to electric drives", *IEEE Trans. Ind. Elec.*, vol 40, n°1, pp.23-36, Feb. 1993.
- [Utk99] V.I. Utkin, A. Sabanovic, "Sliding modes applications in power electronics and motion control systems", *IEEE Conf., ISIE'99*, Bled, Slovenia, pp.TU22-TU31, 1999.
- [Vid89] M. Vidyasagar, M.W. Spong, "Robot dynamics and Control", *John Wiley & Sons*, New York, 1989.
- [Yoo96] B.D. Yoon, Y.H. Kim, H.W. Rhew, C.K. Kim, "An approach for the design of fuzzy sliding adaptive controller in induction motor drives", *IEEE IECON*, vol.2, pp.971-976, 1996.
- [You78] K.-K.D. Young, "Controller design for a manipulator using theory for variable structure systems", *IEEE Trans. Syst. Man. Cybernetics*, vol.8, pp.210-218, 1978.
- [You98] T. Youcef, "Commande à structure variable généralisée appliquée à un bras manipulateur", *Thèse de Magister, INHC Boumerdes*, Algérie, 1998.
- [You99] K.D. Young, V.I. Utkin, Ü. Özgüner, "A control engineer's guide to sliding mode control", *IEEE Trans. on Control Systems Technology*, vol.7, n°3, may 1999.
- [Yu98] Z. Yu, D. Figoli, "AC induction motor control using constant V/Hz principle and space vector PWM technique with TMS320C240", *Application report: SPRA284A, DSP Solutions, Texas Instruments*, April 1998.
- [Zad65] L.A. Zadeh, "Fuzzy Sets", *Information and Control*, vol.8, pp.338-353, 1965.
- [Zad75] L.A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning", Part I, *Information Sciences*, vol.8, pp.301-357, 1975.

Annexe

ANNEXE A. PARAMETRES DU MOTEUR ASYNCHRONE

La simulation est effectuée sur une machine définie par les paramètres suivants :

Puissance nominale : 1,5 kW

Tension nominale : 220 / 380 V

Rendement nominal : 0,78

Facteur de puissance nominal : 0,8

Vitesse nominale : 1420 tr/min

Fréquence nominale : 50 Hz

Résistance rotorique : 3,805 Ω

Résistance statorique : 4,85 Ω

Inductance cyclique du stator : 0,274 H

Inductance cyclique du rotor : 0,274 H

Inductance mutuelle cyclique : 0,258 H

Nombre de paires de pôles : 2

Moment d'inertie : 0,031 Nm·s²/rad

Coefficient de frottement : 0,008 N·m·s/rad

ANNEXE B. CARACTÉRISTIQUES DES ENSEMBLES FLOUS

B.1. Fonctions d'appartenance usuelles

Plusieurs fonctions d'appartenance sont employées dans les systèmes flous. On cite, ci-dessous, les expressions de quelques fonctions fréquemment utilisées. Leurs formes sont illustrées par la figure B.1.

– Les fonctions *triangulaire* et *trapézoïdale* qui sont les plus fréquemment utilisées :

$$\text{trimf}_A(x; [a \ b \ c]) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \quad (\text{B.1})$$

$$\text{trapmf}_A(x; [a \ b \ c \ d]) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \quad (\text{B.2})$$

– La fonction *gaussienne* de probabilité utilisée dans les problèmes nécessitant une adaptation de ses paramètres :

$$\text{gaussmf}_A(x; [\sigma \ x_0]) = \exp\left(-\frac{(x-x_0)^2}{\sigma^2}\right) \quad (\text{B.3})$$

– La fonction en *forme de cloche*, définie par :

$$\text{bellmf}_A(x; [a \ x_0]) = \frac{1}{1 + \left(\frac{x-x_0}{a}\right)^2} \quad (\text{B.4})$$

– La fonction en *forme de cloche généralisée*, qui est une extension de la fonction de distribution de probabilité de Cauchy [Mat99], définie par :

$$\text{gbellmf}_A(x; [a \ b \ x_0]) = \frac{1}{1 + \left(\frac{x-x_0}{a}\right)^{2b}} \quad (\text{B.5})$$

Les paramètres x_0 et a assurent l'ajustement du centre et de la largeur de la fonction d'appartenance, tandis que b agit sur les pentes aux points $(x_0 \pm a, \frac{1}{2})$.

– La fonction *trigonométrique*, définie par :

$$\text{trig}_A(x; [a \ x_0]) = \frac{1}{2} \left(1 + \cos\left(\frac{\pi(x-x_0)}{2a}\right) \right) ; \quad x_0 - 2a \leq x \leq x_0 + 2a \quad (\text{B.6})$$

– La fonction sigmoïdale :

$$\text{smf}_A(x; [a, x_0]) = \frac{1}{1 + \exp(a(x - x_0))} \quad (\text{B.7})$$

La fonction sigmoïdale est souvent utilisée comme fonction d'activation dans les systèmes neuronaux artificiels [Jan92].

– Les fonctions en forme de S :

$$\text{smf}_A(x; [x_0, x_1]) = \begin{cases} 2 \left(\frac{x - x_0}{x_1 - x_0} \right)^2 & x_0 < x < \frac{x_0 + x_1}{2} \\ 1 - 2 \left(\frac{x_1 - x}{x_1 - x_0} \right)^2 & \frac{x_0 + x_1}{2} < x < x_1 \\ 0 & x \leq x_0 \\ 1 & x \geq x_1 \end{cases} \quad (\text{B.8})$$

où x_0 et x_1 déterminent les points d'arrêt de la fonction d'appartenance.

– Les fonctions en forme de Z :

$$\text{zmf}_A(x; [x_0, x_1]) = \begin{cases} 1 - 2 \left(\frac{x - x_0}{x_1 - x_0} \right)^2 & x_0 < x < \frac{x_0 + x_1}{2} \\ 2 \left(\frac{x_1 - x}{x_1 - x_0} \right)^2 & \frac{x_0 + x_1}{2} < x < x_1 \\ 1 & x \leq x_0 \\ 0 & x \geq x_1 \end{cases} \quad (\text{B.9})$$

– La fonction π caractérisée par le produit ou le minimum entre les fonctions en forme de s et en forme de z.

$$\pi \text{mf}_A(x; [x_{10}, x_{11}, x_{20}, x_{21}]) = \text{smf}_A(x; [x_{10}, x_{11}]) \cdot \text{zmf}_A(x; [x_{20}, x_{21}]) \quad (\text{B.10})$$

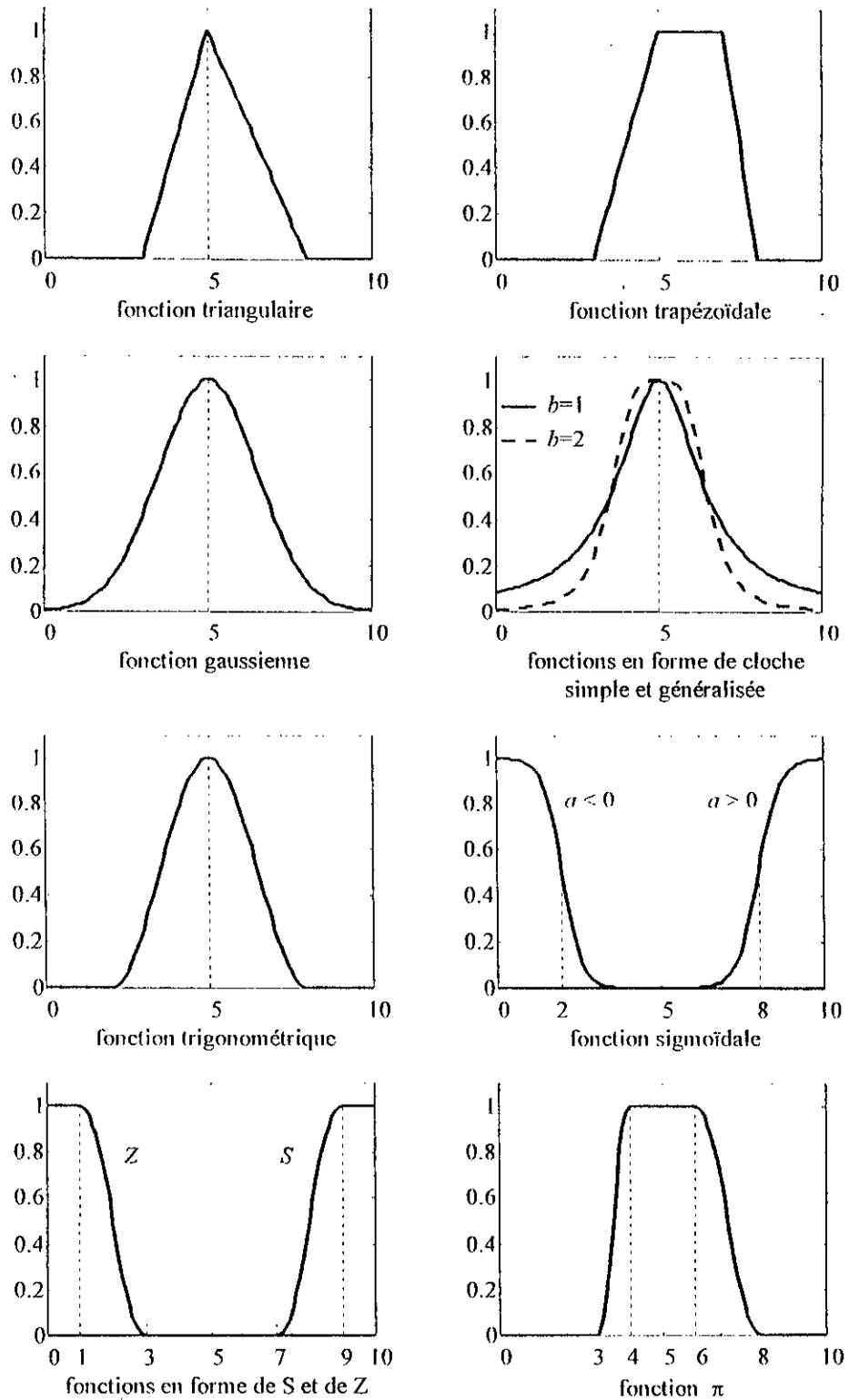


Figure B.1 : Formes usuelles des fonctions d'appartenance

B.2. Opérateurs usuels de conjonction, disjonction et de complément

Sòient A et B deux ensembles flous dans X définis par leurs fonctions d'appartenance respectives $\mu_A(x)$ et $\mu_B(x)$.

– Les degrés d'appartenance de l'intersection et de l'union des ensembles flous A et B sont représentés par une opération de conjonction $T(\cdot, \cdot)$ et de disjonction $S(\cdot, \cdot)$ respectivement :

$$\begin{aligned}\mu_{A \cap B}(x) &= T(\mu_A(x), \mu_B(x)) \\ \mu_{A \cup B}(x) &= S(\mu_A(x), \mu_B(x))\end{aligned}\tag{B.11}$$

avec $T(\cdot, \cdot)$ satisfait les conditions d'une T -norme [Jan92] définies par :

$$\begin{aligned}T(0,0) &= 0 \\ T(a,1) &= T(1,a) = a \\ T(a,b) &\leq T(c,d) \text{ si } a \leq c \text{ et } b \leq d \text{ (monotonie)} \\ T(a,b) &= T(b,a) \text{ (commutativité)} \\ T(a,T(b,c)) &= T(T(a,b),c) \text{ (associativité)}\end{aligned}\tag{B.12}$$

et $S(\cdot, \cdot)$ satisfait les conditions d'une T -conorme définies par :

$$\begin{aligned}S(1,1) &= 1 \\ S(a,0) &= S(0,a) = a \\ S(a,b) &\leq S(c,d) \text{ si } a \leq c \text{ et } b \leq d \text{ (monotonie)} \\ S(a,b) &= S(b,a) \text{ (commutativité)} \\ S(a,S(b,c)) &= S(S(a,b),c) \text{ (associativité)}\end{aligned}\tag{B.13}$$

– Le degré d'appartenance de \bar{A} (la *négation* de A) est spécifié par un opérateur de négation $N(\cdot)$ satisfaisant la condition :

$$\begin{aligned}N(0) &= 1 \text{ et } N(1) = 0 \\ N(a) &\leq N(b) \text{ si } a \geq b \text{ (monotonie)}\end{aligned}\tag{B.14}$$

Remarque :

La T -norme $T(\cdot, \cdot)$ et la T -conorme $S(\cdot, \cdot)$ doivent vérifier la généralisation de la loi de De Morgan, qui est donnée par [Kel97] :

loi de De Morgan généralisée

$$S(a,b) = N[T(N(a), N(b))]$$

$$T(a,b) = N[S(N(a), N(b))]$$

Loi de De Morgan

$$A \cup B = \overline{\overline{A} \cap \overline{B}}$$

$$A \cap B = \overline{\overline{A} \cup \overline{B}}$$

(B.15)

B.3. Quelques exemples des opérateurs de T-norme, de T-conorme et de complément

Le tableau B.1 montre quelques opérateurs de T-norme, de son conjoint T-conorme et du complément, utilisés dans la littérature [Bou95].

Nom	T-norme $T(a,b)$	T-conorme $S(a,b)$	Complément $N(a)$
Zadeh	$\min(a, b)$	$\text{Max}(a, b)$	$1 - a$
Probabiliste	$a \cdot b$	$a + b - a \cdot b$	$1 - a$
Lukasiewicz	$\max(a + b - 1, 0)$	$\min(a + b, 1)$	$1 - a$
Hamacher ($\gamma > 0$)	$\frac{a \cdot b}{\gamma + (1 - \gamma)(a + b - ab)}$	$\frac{a + b + a \cdot b - (1 - \gamma) \cdot ab}{1 - (1 - \gamma) \cdot ab}$	$1 - a$
Yager ($p > 0$)	$\max\left(1 - \left((1 - a)^p + (1 - b)^p\right)^{1/p}, 0\right)$	$\min\left(\left(a^p + b^p\right)^{1/p}, 1\right)$	$1 - a$
Weber ($\lambda > -1$)	$\max\left(\frac{a + b - 1 + \lambda \cdot ab}{1 + \lambda}, 0\right)$	$\min(a + b + \lambda \cdot ab, 1)$	$\frac{1 - a}{1 + \lambda a}$
Drastique	a si $b = 1$ b si $a = 1$ 0 sinon	a si $b = 0$ b si $a = 0$ 1 sinon	$1 - a$

Tableau B.1 : Exemple d'opérateurs de conjonction, disjonction et de complément
(Cf. [Bou95], p. 41)

ANNEXE C. DEMONSTRATION DU LEMME (4.1)

Soit \tilde{N} est un ensemble flou dont le support est strictement négatif, c'est à dire :

$$\forall z \in \mathbb{R}^+ : \mu_{\tilde{N}}(z) = 0 \quad (\text{C.16})$$

\tilde{e} le vecteur d'erreur flou, $\tilde{s}(\tilde{e})$ le vecteur flou de commutation défini par :

$$\tilde{s}(\tilde{e}) = \begin{pmatrix} \tilde{s}_1(\tilde{e}) \\ \dots \\ \tilde{s}_m(\tilde{e}) \end{pmatrix} \quad (\text{C.17})$$

Lemme : La condition d'atteignabilité de la dynamique correspondant à la $i^{\text{ème}}$ entrée, définie par :

$$\exists \tilde{N} \text{ tel que } \tilde{s}_i(\tilde{e}) \cdot \dot{\tilde{s}}_i(\tilde{e}) \subset \tilde{N} ; i = 1, \dots, m \quad (\text{C.18})$$

est équivalente à la condition (C.19) définie par :

$$\forall s_i(e) \in \text{Supp}(\tilde{s}_i(\tilde{e})) - \{0\} \text{ et } \forall \dot{s}_i(e) \in \text{Supp}(\dot{\tilde{s}}_i(\tilde{e})) - \{0\} \quad (\text{C.19})$$

Alors : $s_i(e) \cdot \dot{s}_i(e) < 0$

Preuve : [Gha97]

Soit un ensemble flou $F_{s\dot{s}}$ défini par :

$$F_{s\dot{s}} = \tilde{s}_i(\tilde{e}) \sim \dot{\tilde{s}}_i(\tilde{e}) \quad (\text{C.20})$$

avec : \sim l'extension de l'opérateur de multiplication sur les ensembles flous.

Le support de $F_{s\dot{s}}$ sera donc décrit par l'expression :

$$\text{supp}(F_{s\dot{s}}) = \{s_i(e) \cdot \dot{s}_i(e) \in \mathbb{R} \mid s_i(e) \in \text{supp}(\tilde{s}_i(\tilde{e})) \text{ et } \dot{s}_i(e) \in \text{supp}(\dot{\tilde{s}}_i(\tilde{e}))\} \quad (\text{C.21})$$

Utilisant l'équation (C.20), la condition (C.18) peut être écrite par :

$$F_{s\dot{s}} \subset \tilde{N} \quad (\text{C.22})$$

Utilisant la définition de l'inclusion des ensembles flous [Zad65], la condition (C.22) devient :

$$\forall z \in \mathbb{R} : \mu_{F_{s\dot{s}}}(z) < \mu_{\tilde{N}}(z) \quad (\text{C.23})$$

Si on prend $z_i = s_i(e) \cdot \dot{s}_i(e) \in \text{supp}(F_{s\dot{s}})$ et $z \geq 0$, les équations (C.23) et (C.16) conduit à :

$$\mu_{F_{s\dot{s}}}(z_i = s_i(e) \cdot \dot{s}_i(e)) < 0 \quad (\text{C.24})$$

Ce qui est une contradiction, car les degrés d'appartenance appartiennent à l'intervalle $[0,1]$. Donc, nous pouvons conclure que :

$$\forall z_i = s_i(e) \cdot \dot{s}_i(e) \in \text{supp}(F_{ss}) : s_i(e) \cdot \dot{s}_i(e) < 0 \quad (\text{C.25})$$

D'autre part, pour que la condition (C.25) soit vérifiée, il faut qu'en plus les vecteur $s_i(e)$ et $\dot{s}_i(e)$ ne soient pas nuls.

Cependant, à partir des équations (C.21), (C.24) et (C.25), nous pouvons conclure que pour tout $s_i(e) \in \text{supp}(\tilde{s}_i(\tilde{e})) - \{0\}$ et $\dot{s}_i(e) \in \text{supp}(\tilde{\dot{s}}_i(\tilde{e})) - \{0\}$ nous avons $s_i(e) \cdot \dot{s}_i(e) < 0$.

ANNEXE D. LES FONCTIONS DE CARATHEODORY

Avant de présenter les fonctions de Carathéodory, quelques définitions de base sont données au préalable [Cha81][Uni98].

D.1.1. Espace topologique

– *Espace* : ensemble muni d'une structure.

– *Espace topologique* : ensemble muni d'une structure topologique.

– *Topologie* (ou structure topologique) sur un ensemble E est un ensemble \mathcal{G} de parties de E satisfaisant aux propriétés suivantes :

- a) la réunion de toute famille d'éléments de \mathcal{G} appartient à \mathcal{G}
- b) l'intersection de toute famille finie d'éléments de \mathcal{G} appartient à \mathcal{G}

En particulier, la partie vide et la partie pleine de E appartiennent à \mathcal{G} .

– On appelle *partie ouverte* (ou un *ouvert*) de l'espace topologique E toute partie de E appartenant à \mathcal{G} , et étant un voisinage de chacun de ses points.

– On appelle *partie fermée* (ou un *fermé*) toute partie de E dont le complémentaire est un ouvert.

– Si E est un espace discret, toute partie de E est à la fois ouverte et fermée.

D.1.2. Espace métrique

– L'*espace métrique* est un ensemble E muni d'une *distance* d [Cha81].

– On appelle *distance* sur un ensemble E une application d de $E \times E$ dans l'ensemble \mathbb{R}_+ (des nombres réels positifs ou nul) telle que, quels que soient les éléments x, y et z de E , on ait:

$$d(x, y) = 0 \Leftrightarrow x = y \quad (\text{D.26a})$$

$$d(x, y) = d(y, x) \quad (\text{D.26b})$$

$$d(x, z) \leq d(x, y) + d(y, z) \quad (\text{D.26c})$$

La dernière condition (D.26c) est appelée *inégalité triangulaire* car elle est la généralisation de la classique inégalité entre les longueurs des côtés d'un triangle.

Tout sous-ensemble A de E est un espace métrique, dit sous-espace métrique de E pour la distance d' définie par $d'(x, y) = d(x, y)$, $\forall x, y \in A$.

Exemple :

\mathbb{R}^2 est un espace métrique muni d'une distance d qui peut prendre au moins l'une des formes suivantes :

$\forall x = (x_0, x_1)$ et $y = (y_0, y_1)$ de \mathbb{R}^2 :

$$d(x,y) = |x_0 - y_0| + |x_1 - y_1|$$

$$d(x,y) = (x_0 - y_0)^2 + (x_1 - y_1)^2 \quad (\text{D.27})$$

$$d(x,y) = [(x_0 - y_0)^2 + (x_1 - y_1)^2]^{1/2}$$

D.1.3. Espace topologique localement compact

- Un espace topologique E est dit *séparé* si, pour tout couple (x,y) de points distincts de E , il existe un voisinage de x et un voisinage de y disjoints. Cette condition est appelée *axiome de Hausdorff*.
- Tout espace métrique est séparé (en particulier \mathbb{R}^n).
- Un *recouvrement* d'une partie P d'un ensemble E est une famille $(P_i)_{i \in I}$ de parties de E dont la réunion contient P .
- Lorsqu'une sous-famille $(P_i)_{i \in J}$ de la famille $(P_i)_{i \in I}$ est un recouvrement de P , on dit que le recouvrement $(P_i)_{i \in J}$ est *extrait* du recouvrement $(P_i)_{i \in I}$.
- Un recouvrement $(P_i)_{i \in I}$ d'une partie P d'un espace topologique E est dit *ouvert* (resp. *fermée*) si, pour tout élément i de I , P_i est un *ouvert* (resp. *fermé*) de E .
- Un *espace topologique* E est *compact* s'il est séparé, et si de tout recouvrement ouvert de E on peut extraire un recouvrement fini.
- Pour qu'un sous-espace d'un espace compact E soit compact, il faut et il suffit qu'il soit fermé dans E .
- Un espace topologique séparé E est dit *localement compact* si tout point de E possède un voisinage compact.
- Tout espace compact est localement compact.
- Les parties ouvertes et les parties fermées d'un espace localement compact sont localement compactes.
- Le produit d'une famille finie d'espaces localement compact est localement compact.

D.1.4. Fonction mesurable

- Soit $P(E)$ ensemble des parties de E . Une *tribu* de parties d'un ensemble E est toute partie non vide de $P(E)$ tels que : la réunion de toute famille d'éléments de $P(E)$ appartient à $P(E)$, et le complément de chaque élément de $P(E)$ appartient à $P(E)$. [Cha81]
- *Espace mesurable* : couple constitué d'un ensemble E et d'une tribu $P(E)$ de parties de E . On définit dans $P(E)$ une application $m : P(E) \rightarrow \mathbb{R}_+$ (appelée *une mesure*) vérifiant la condition [Rud64] [Uni98] :

$$\forall A, B \in P(E) \text{ alors } A, B \text{ disjoints} \Rightarrow m(A \cup B) = m(A) + m(B) \quad (\text{D.28})$$

- Les parties de E appartenant à une tribu $P(E)$ sont dites des *ensembles mesurables*.
- *fonction mesurable* : Soit f une fonction à valeurs réelles définie sur un espace mesuré E . La fonction f est dite *mesurable* si l'ensemble $\{x / f(x) > a\}$ est mesurable pour tout réel a [Rud64].

D.1.5. Mesure au sens de Lebesgue

L'application $f \rightarrow \mu(f) = \int f$ est une mesure de Radon positive sur \mathbb{R} , appelée *mesure de Lebesgue* et notée m_x [Cha81].

On appelle mesure de Lebesgue sur \mathbb{R}^n la mesure de Radon :

$$m_x = m_{x_1} \otimes m_{x_2} \otimes \dots \otimes m_{x_n} \quad (\text{D.29})$$

où, pour tout élément i de $[1, n]$, m_{x_i} désigne la mesure de Lebesgue sur \mathbb{R} .

D.1.6. Mesure au sens de Radon

Soit : E un espace topologique localement compact dénombrable à l'infini (exemple \mathbb{R}^n), et $KC(E)$ l'espace vectoriel des fonctions à valeurs réelles continues sur E à support compact.

On appelle mesure de Radon sur E une forme linéaire $\mu : KC(E) \rightarrow \mathbb{C}$ satisfaisant à la condition suivante :

$$\begin{aligned} &\text{Pour toute fonction positive } f \text{ appartenant à } KC(\mathbb{R}), \\ &\text{il existe un nombre réel positif } M \text{ tel que,} \\ &\forall g \text{ de } KC(E) : |g| < f \Rightarrow |\mu(g)| \leq M. \end{aligned} \quad (\text{D.30})$$

Toute forme linéaire positive sur $KC(E)$ est une mesure de Radon, dite positive.

D.1.7. Fonction fortement Carathéodory

Une fonction $f(\cdot) : U_x \times \mathbb{R} \rightarrow \mathbb{R}^n$ est une *fonction fortement Carathéodory* [Gha97] si, et seulement si, pour tout $t \in \mathbb{R}$, la fonction $f(\cdot, t)$ est continue ;

pour tout $x \in U_x$, $f(x, \cdot)$ est mesurable au sens de Lebesgue ;
et pour tout ensemble compact Σ de $U_x \times \mathbb{R}$, il existe une constante réelle C_Σ telle que,
pour tout (x, t) de Σ on a :

$$\|f(x, t)\| \leq C_\Sigma \quad (\text{D.31})$$

Résumé

La commande des systèmes complexes et incertains, entre autres les moteurs asynchrones, est un problème contrariant pour les chercheurs. De ce fait, plusieurs stratégies de commande linéaire et non-linéaire ont vu le jour pour résoudre les problèmes de découplage, de robustesse, de simplicité de réglage et de mise en œuvre. La commande par logique floue, vu sa capacité, constitue une bonne solution pour ce genre de problèmes.

Dans ce contexte, nous avons étudié différents aspects de la logique floue, des contrôleurs flous et des modèles flous, ainsi que leur introduction dans la commande des systèmes. Nous avons axé notre travail sur une technique de modélisation floue, servant à imiter le système par une représentation floue tout en tenant compte de la description analytique grossière et des incertitudes du système. Deux commandes ont tiré profit de cette technique :

- ♦ Une commande par flux orienté basée sur un modèle inverse flou,
- ♦ Une commande floue robuste à structure variable basée sur un modèle flou du moteur.

Mots clés : Modélisation floue, commande basée sur un modèle flou, retour d'état non-linéaire, commande par flux orienté, commande à structure variable, commande à structure variable floue, modèle inverse flou.

Abstract

The control of complex and uncertain dynamical systems, particularly the induction motor drives, is a problem for the researchers. Thus, several linear and non-linear control strategies have appeared to solve the decoupling, robustness and practical problems. Fuzzy logic control is a good solution to such problems.

In this field, we have studied different aspects of fuzzy logic, fuzzy controllers and fuzzy models, even their introduction in control systems. We have pointed our work in a technique of fuzzy modelling, which help to imitate the system model by a linguistic fuzzy rule base representation, with taking into account the mathematical description and the boundary uncertainties of the system. Two control systems have profited from this technique:

- ♦ Control based on an inverse fuzzy model with field orientation,
- ♦ Fuzzy robust control based on a linguistic fuzzy model of the motor.

Key words: Fuzzy modelling, fuzzy model-based control, non-linear state feedback, field oriented control, variable structure control, fuzzy variable structure control, inverse fuzzy model.

ملخص

إن التحكم في الأنظمة المعقدة وغير الدقيقة، لاسيما المحركات الكهربائية اللامتزامنة الثلاثية الأطوار، يعتبر مشكلة تواجه الباحثين. وإنطلاقاً من ذلك، برزت عدة إستراتيجيات للتحكم الخطي وغير الخطي من أجل حل مشاكل الفصل بين المتغيرات الداخلية، وعدم التأثير بعوامل الاضطرابات والإرتيابات، وبساطة الضبط والتشغيل. ويأتي التحكم بواسطة المنطق الغامض كحل جيد لمثل هذه المشاكل، وذلك بفضل قدراته الخاصة.

وفي هذا الإطار، قمنا بدراسة عدة جوانب للمنطق والمراقبات والنماذج الغامضة، بالإضافة إلى إدخال هذه الأخيرة على أنظمة التحكم. وقد زكنا عملنا على تقنية لتشكيل النماذج الغامضة، تسمح بتمثيل غامض للنظام أخذاً بعين الاعتبار وصفه التحليلي (الرياضي) العام وإرتيابه. حيث طبقنا هذه التقنية في إعداد شكلين من التحكم، هما:

- ♦ تحكم بتدفق موجه قائم على نموذج عكسي غامض،
- ♦ تحكم غامض غير متأثر بالاضطرابات والإرتيابات وذات هيكل متغير قائم على نموذج غامض للنظام.

العلامات المفتاحية: تشكيل غامض للنماذج، تحكم قائم على نموذج غامض، تغذية إسترجاعية غير خطية، تحكم بتدفق موجه، تحكم ذات هيكل متغير، تحكم غامض ذات هيكل متغير، نموذج عكسي غامض.