

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique



Thèse

En vue de l'obtention du titre de
Docteur en Sciences
en Automatique par

SID-ALI AMAMRA

Magister en Automatique – ENP

Modélisation Inverse Floue des Systèmes Incertains Application : Machine Asynchrone Implantation sur une carte dSPACE

Soutenu publiquement le Mercredi 02 Novembre 2011, devant le jury composé de :

PRESIDENT:

TADJINE MOHAMED, PROF, ENP

DIRECTEURS:

BOUCHERIT MOHAMED SEGHIR, PR., ENP

BARAZANE LINDA, PR., USTHB

CHERIFI ABDEREZZAK, PR., UVSQ, IUT MANTES LA JOLIE

EXAMINATEURS:

NEZLI LAZHARI, MCA, ENP

LARABI ABDELKADER, MCA, USTHB

BENMANSOUR KHELIFA, MCA, UYFM, MEDEA

LABIOD SALIM, MCA, UNIVERSITE DE JIJEL

Année : 2010 - 2011

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*C'est
À mes parents, frères et sœurs
À tous mes amis et tous ceux qui m'aiment,
Aux chers lecteurs
Que je dédie ce modeste travail*

AVANT-PROPOS

Cette thèse est le fruit de beaucoup d'efforts de recherches passionnantes, mais aussi très prenantes. J'espère que ceci ait un bon acquis à ceux qui veulent se performer et développer les résultats obtenus.

Le travail de recherche de cette thèse a été effectué en collaboration entre le Laboratoire de Commande de Processus (LCP) de l'Ecole Nationale Polytechnique d'Alger, et le Laboratoire d'Ingénierie des Systèmes de Versailles (LISV) de l'Université de Versailles Saint Quentin en Yvelines, site de Mantes-la-Jolie.

Mes pensées emplumées de mercis voltigent majestueusement vers mes compagnons de fortune, qui sont mes superviseurs de thèse, Prof. M.S. Boucherit, Prof L. Barazane et Prof. A. Cherifi Directeurs de thèse, pour m'avoir fourni ce sujet de recherche, pour leurs suivis rigoureux, leurs directives et remarques pertinentes et leurs aides et soutiens tout au long du déroulement de ce travail.

Ma reconnaissance et mes remerciements sont dus au Professeur M. Tadjine pour avoir accepté d'être président de jury. Je remercie également M. L. Nezli, M. A. Larabi, M. K. Benmensour et M. S. Labiod pour avoir accepté d'être membres examinateurs de jury.

Des remerciements spéciaux vont aux chercheurs du LCP de l'ENP, pour leur gentillesse et accueil chaleureux. Je tiens également à remercier tous les professeurs du département Génie Electrique.

Mes ultimes et tendres pensées vont à mes proches amis, et surtout à ma famille adorée et particulièrement à mes parents, sans qui cette aventure n'aurait pas été possible, car leurs prières, dans les moments les plus difficiles, n'ont cessé de me porter.

Table de Matières

Notations	8
Introduction Générale	9
1. Chapitre 1 : Contribution à la modélisation floue des systèmes incertains ...	14
1.1 Introduction.....	15
1.2 Evaluation des Systèmes réels et flous.....	16
1.3 Les modèles flous à base de règles.....	17
1.3.1 Partitionnement flou de l'espace d'entrée	18
1.3.1.1 Partitionnement homogène.....	18
1.3.1.2 Partitionnement par concaténation d'ensembles flous.....	20
1.3.1.3 Partitionnement optimal.....	21
1.3.2 Différents types de modèles flous.....	21
1.3.2.1 Le modèle flou linguistique.....	22
1.3.2.2 Modèle flou de Takagi-Gugeno.....	28
1.3.2.3 Modèle flou de Tsukamoto.....	31
1.4 Elaboration d'un modèle flou.....	32
1.5 Modélisation floue linguistique des systèmes incertains.....	33
1.5.1 Condition d'existence de modèle flou selon la méthode proposée....	34
1.5.2 Algorithme proposé de la modélisation floue.....	35
1.5.3 Evaluation de la base de règles du vecteur de commande.....	40
1.5.4 Obtention du modèle flou.....	41
1.6 Modèle inverse flou.....	42
1.6.1 Elaboration du modèle inverse flou par l'inversion d'une base de règles.....	43
1.6.2 Elaboration du modèle inverse flou à partir d'une connaissance analytique du modèle inverse.....	46
1.7 Conclusion.....	47
2. Chapitre 2 : Application du MIF à la commande de la machine asynchrone	48
2.1 Introduction.....	49
2.2 Contrôle classique de la MAS.....	50
2.2.1 Introduction.....	51
2.2.2 Modèle de la machine asynchrone.....	51
2.2.2.1 Hypothèse de travail.....	51
2.2.2.2 Les équations de la machine asynchrone en régime quelconque..	51
2.2.3 Régulation, méthode classique.....	56
2.3 Contrôle par modèle inverse flou « MIF »	60

2.3.1	Introduction.....	60
2.3.2	Modèle inverse flou.....	60
2.4	Etude de Robustesse.....	70
2.4.1	Introduction.....	70
2.4.2	Paramètres mécaniques.....	70
2.4.2.1	Moment d'inertie (J).....	70
2.4.3	Paramètres électriques.....	73
2.4.3.1	Résistance rotorique (Rr).....	73
2.4.3.2	Inductance rotorique (Lr).....	74
2.5	Conclusion.....	76
3.	Chapitre 3 : Application des systèmes évolués au modèle inverse flou....	77
3.1	Introduction.....	78
3.2	Commande IP par modèle inverse flou appliquée au MAS.....	78
3.3	Commande à structure variables floue de la machine asynchrone basée sur le modèle inverse flou.....	83
3.3.1	Algorithme de floue.....	85
3.3.2	Application à la commande du moteur asynchrone basée sur le modèle inverse flou.....	90
3.3.2.1	Introduction.....	90
3.3.2.2	Elaboration de la commande floue robuste.....	91
3.4	Commande adaptative floue de la machine asynchrone basée sur le modèle inverse flou.....	96
3.4.1	Introduction.....	96
3.4.2	Structure de commande proposée.....	99
3.4.3	Application au modèle inverse flou.....	104
3.5	conclusion.....	107
4.	Chapitre 4 : Dispositif expérimental.....	108
4.1	Introduction.....	109
4.2	Identification des paramètres de la machine asynchrone.....	109
4.2.1	Mesure de la résistance statorique.....	111
4.2.2	Essai à vide.....	111
4.2.3	Détermination des paramètres mécaniques.....	112
4.2.4	Essai en court-circuit.....	113
4.2.5	Validation du modèle.....	115
4.3	description générale.....	118
4.3.1	Partie puissance.....	118
4.3.1.1	Prototype MAS rotatif.....	118
4.3.1.2	Prototype convertisseur.....	119
4.3.1.3	Description de l'IGBT.....	120
4.3.2	Partie commande.....	124
4.3.2.1	Carte de commande dSPACE – DS 1103.....	124

4.3.2.2 Drivers d'IGBT.....	126
4.3.2.3 Circuit d'adaptation des signaux de commande.....	130
4.3.2.4 Capteur de vitesse et de couple de rotation.....	130
4.3.3 Partie software.....	131
4.4 Structure générale de banc expérimental.....	133
4.5 Conclusion.....	137
Conclusion Générale.....	138
Suggestions et perspectives.....	139
Annexe.....	140
A. Caractéristiques des ensembles flous.....	141
B. Terminologie de la logique flou type-2.....	146
C. Démonstration du Lemme (3.1)	149
D. Les fonctions de Carathéodory.....	151
Références bibliographique.....	155

Notations

R_r	Résistance rotorique par phase
R_s	Résistance statorique par phase
L_r	Inductance rotorique par phase
L_s	Inductance statorique par phase
L	Inductance du filtre
M_{sr}	Inductance mutuelle
a, b, c	Indices correspondants aux trois phases du stator
A, B, C	Indices correspondants aux trois phases du rotor
T_r	Constante de temps rotorique $T_r=L_r/R_r$
T_s	Constante de temps statorique $T_s=L_s/R_s$
σ	Coefficient de dispersion
J	Moment d'inertie de la partie tournante
f	Coefficient de frottement visqueux
p	Nombre de paires de pôles
$d-q$	Axes correspondant au référentiel lié au champ tournant
Φ_{dr}	Flux rotorique sur l'axe d
Φ_{qr}	Flux rotorique sur l'axe q
Φ_r	Flux rotorique résultant
I_{ds}	Courant statorique sur l'axe d
I_{qs}	Courant statorique sur l'axe q
θ_s	Angle électrique statorique
ω_s	Vitesse angulaire électrique statorique
ω_r	Vitesse angulaire électrique rotorique
ω_{sl}	Glissement de vitesse angulaire électrique
C_{em}	Couple électromagnétique
C_r	Couple résistant
C_n	Couple nominal
Ω_r	Vitesse mécanique
$\Omega_{r \text{ réf}}$	Vitesse mécanique de référence

Introduction Générale

Bien que déjà présente dans d'autres domaines, la logique floue et plus précisément la modélisation floue constitue, au début de notre travail de thèse, une nouveauté dans le domaine de commande des machines électrique en général, et le moteur à induction en particulier [1,2,3,4,5,6,7 et 8]. Nous avons voulu savoir quel pourrait être l'apport de cette méthode appliquée à la commande de la machine asynchrone. Des questions se posent alors naturellement : cette méthode qui ne relève, en tout cas pas à première vue, d'une logique cartésienne classique, peut-elle conduire à de meilleurs résultats ? Sinon, que présente-t-elle comme avantages et inconvénients par rapport aux techniques conventionnelles ?

L'objectif de la modélisation est d'obtenir un modèle formel permettant de décrire un processus naturel, humain ou industriel en vue de sa compréhension (e.g. les moteurs électriques tournants en général), et/ou en vue de la prédiction des effets de ce processus. Fondée sur le concept de règle floue [8], l'approche floue propose une modélisation plus intuitive que les approches classiques de modélisation, qu'elles soient probabilistes ou analytiques, basées sur des systèmes d'équations différentielles ou de dérivées partielles. En tant que brique élémentaire, une règle floue permet de modéliser une connaissance experte, en reliant de façon non linéaire des entrées floues avec des sorties nettes ou floues. Héritée de l'approche symbolique de l'Intelligence Artificielle, la méthode classique de modélisation par système flou consiste à faire expliciter linguistiquement par l'expert son processus de décision afin d'obtenir les règles floues nécessaires [9,10,11,12,13,14 et 15]. Cependant, dans de nombreux cas (processus difficilement modélisables par des méthodes classiques, grande complexité du processus de décision, processus incertains), il est à priori impossible d'interpréter la prise de décision sous la forme de règles linguistiques [14 et 15]. L'approche Ben-Ghali propose dans ces cas d'extraire automatiquement, par apprentissage, les règles de décision représentant le processus à modéliser. Cet apprentissage s'effectue à l'aide d'une base de données d'exemples comportant diverses prises de décision en fonction de différentes situations.

Comme tout système flou se révèle être un approximeur universel de fonctions non linéaires et incertains reliant les variables de sortie aux variables observées [14,15 et 16], il permet théoriquement de modéliser n'importe quel processus ou système dynamique non linéaire complexe, faisant intervenir un grand nombre de variables en forte interaction (cas de la machine asynchrone). Tout modèle flou bénéficie des quatre propriétés importantes des systèmes flous [14,15,16,17,18,19,20,21 et 22] à savoir Intelligibilité, la Localité, la Traçabilité et la Granularité.

A la différence de l'approche connexionniste (i.e. réseaux de neurones formels) qui construit des modèles de type boîte noire [13,23 et 24], l'induction automatique de modèles flous permet de maintenir les propriétés d'intelligibilité, de traçabilité, de localité et de granularité qui font la force des systèmes flous, tout en offrant une grande souplesse dans la définition de modèles non linéaires, même en l'absence de connaissance a priori.

La commande à flux orienté a montré son efficacité dans le découplage entre les variables (couple et flux) en régime permanent, cependant il s'est avéré que ce type de commande reste sensible aux incertitudes inhérentes au système, surtout lorsqu'il s'agit des variations paramétriques et des perturbations externes. Dans ce contexte, nous avons présenté une nouvelle approche de cette commande à flux orienté en utilisant la technique intelligente de la modélisation floue, sans toucher au régulateur classique, nous avons contribué à la modélisation inverse floue du moteur à induction, afin de concevoir un modèle inverse flou susceptible d'englober toutes les incertitudes du moteur et d'approximer le modèle inverse réel et de fournir une commande robuste tout en gardant la propriété d'orientation du flux rotorique.

Le fruit de notre travail de doctorat s'est concrétisé par une contribution à la modélisation inverse floue de la machine asynchrone afin d'élaborer une commande à flux orienté robuste basée sur le modèle inverse flou de la machine et ce sans toucher au régulateur classique. Une fois cette technique est appliquée, le système global est susceptible de fournir une commande robuste floue qui prend en charge toutes les incertitudes. L'approche proposée offre aussi l'avantage de conserver les performances de la commande classique par flux orienté sans modifier le régulateur classique.

Tout au long de ce travail, nous avons gardé comme objectif l'implantation et l'expérimentation des méthodes développées. L'aspect expérimental a une très grande importance. Il permet de valider les méthodes établies par un travail théorique et de simulation. Il met également en évidence certaines lacunes dues à une mauvaise identification. Dans de nombreux cas, les résultats sont assez satisfaisants puisque la modélisation floue pallie ces erreurs. L'expérimentation en laboratoire constitue donc un garde-fou avant l'étape d'industrialisation. Cette thèse porte essentiellement sur deux axes ; Développement d'un nouveau algorithme de commande des systèmes incertains basé sur leurs modèles inverse flous appliqué au moteur à induction et Développement des structures de commandes non linéaires basées sur le modèle inverse flou. Nous avons orienté notre travail vers la modélisation inverse floue des systèmes incertains dans le but d'élaborer une nouvelle approche de commande robuste à flux orienté basée sur le modèle inverse flou de la machine

à induction. Afin de bénéficier de cet avantage apporté par le modèle inverse flou, et d'augmenter la robustesse vis-à-vis de la variation paramétrique de la machine, d'autres techniques de commande à savoir par un régulateur IP anti-saturation, mode glissant ou bien adaptative toutes basées sur la logique floue ont été développées. Une plateforme expérimentale, pilotée par une carte dSPACE DS 1103, a été développée afin d'être utilisée dans les tests de validation expérimentale des différents résultats obtenus en simulation.

Les différents travaux concernant le sujet font l'objet de quatre chapitres qui constituent ce mémoire.

Le chapitre 1 présente la nouvelle approche de cette étude. Cette technique étant relativement nouvelle dans la communauté de génie électrique, il est important de bien préciser les termes employés et de développer suffisamment la méthode utilisée. Dans cette partie l'accent est mis aussi bien sur les étapes et l'algorithme de conception du modèle que sur leur utilisation. Ceci afin de se familiariser avec une manière de raisonner différente par rapport à l'approche à laquelle on est habitué dans la résolution de certains problèmes.

L'application de cette nouvelle technique à la commande du moteur à induction fait l'objet du chapitre 2. En premier lieu, nous présentons brièvement l'approche classique de commande de la machine asynchrone par orientation du flux rotorique. Par la suite, un modèle intelligent basé sur le modèle mathématique de la machine en tenant compte de la condition d'orientation du flux rotorique et en utilisant la technique de la modélisation floue pour la commande du moteur est présenté. Une comparaison entre les deux méthodes est abordée de manière plus détaillée en présentant une étude, par simulation, sur l'influence des changements de paramètres de la machine sur la réponse du système.

Dans le chapitre 3, nous essayons de tirer d'avantage de la technique de la modélisation floue appliquée à la commande de la machine asynchrone et de l'appliquer avec des régulateurs plus performants que le régulateur classique PID. Dans un premier lieu un régulateur IP est présenté afin de pallier au problème de blocage par saturation de la sortie intégrale du régulateur classique proportionnel intégrale [9], ensuite un régulateur par mode glissant flou [4] est appliqué au modèle inverse flou pour la commande du moteur à induction et finalement une technique de commande adaptative floue [25] est aussi appliquée au moteur, mais ces dernières techniques ont été prouvées juste par la simulation numérique et ce à cause de la limite de la taille mémoire de la carte dSPACE utilisée au cours de cette étude.

Chapitre 4, présente dans un premier temps, l'identification des paramètres de moteur asynchrone par les différents essais de démarrage à vide et à rotor bloqué avec la validation du modèle par la simulation et l'expérimentation, après une présentation du banc d'essai est présentée.

Nous terminons notre travail par une conclusion sur l'ensemble de cette étude et nous proposons des perspectives de travail.

Chapitre 1

Chapitre 1

Contribution à la Modélisation Floue des Systèmes Incertains

1.1. Introduction

L'approche traditionnelle de la modélisation est basée sur une parfaite connaissance du système à étudier [14 et 27], ainsi que sur des connaissances mathématiques appropriées au développement du modèle [18,19,28 et 29].

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 [1,3,5,6 et 28]. 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 incertains avec un certain degré d'exactitude. Comparé aux autres techniques intelligentes 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 [16,29,30,31 et 32].

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.

1.2. Évaluation des systèmes réels et flous

On distingue trois différents types de systèmes [16,31,32 et 33] :

- les systèmes décrits par des fonctions réelles, qui sont généralement des systèmes certains (figure 1.1).

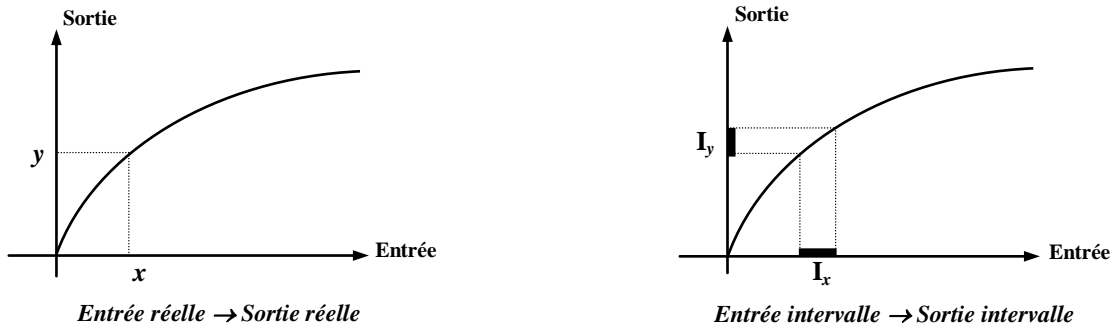


Figure 1.1 : Évaluation d'une fonction réelle pour des arguments réels

- les systèmes décrits par des fonctions intervalles (figure 1.2).

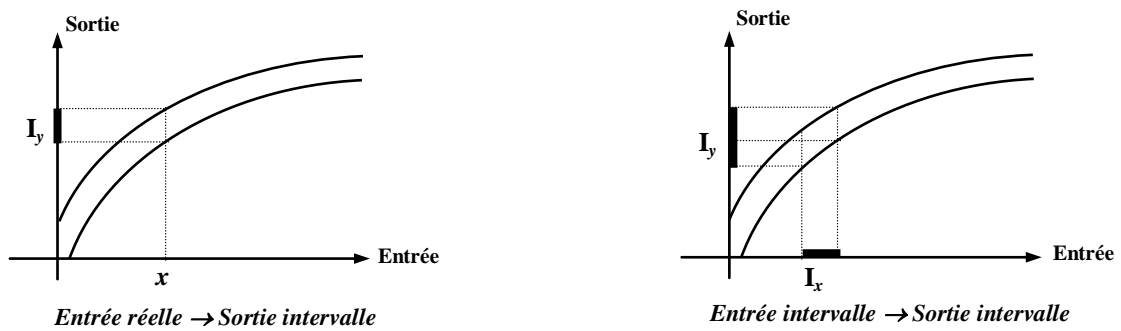


Figure 1.2 : Évaluation d'une fonction intervalle pour des arguments intervalles

- les systèmes décrits par des fonctions floues (figure 1.3).

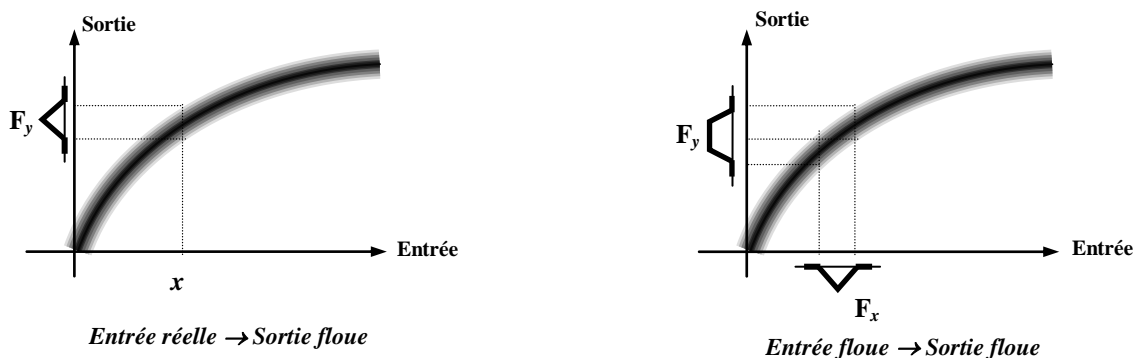


Figure 1.3 : Évaluation d'une fonction floue pour des arguments flous

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.

1.3. 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 [10,11 et 12]:

$$\mathbf{IF} \ \tilde{x} \ \mathbf{is} \ F_x \ \mathbf{THEN} \ \text{consequence} \quad 1.1$$

\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 multivariable* $\mu_x(\mathbf{x})$ donnée dans l'intervalle [0,1] comme suit :

$$\mu_x(x) : x \in \mathbb{R}^n \rightarrow [0,1] \quad 1.2$$

Dans des cas particuliers, cette fonction d'appartenance multivariable peut être déterminée par des combinaisons logiques (min, max, prod, etc...) 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 [7,8 et 14].

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 :

$$\mu_{F_x}(x) = \exp\left(\frac{-(x_1^2 + x_2^2)}{\sigma^2}\right) \quad 1.3$$

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

$$\mu_{F_{x_1}}(x_1) = \exp\left(\frac{-x_1^2}{\sigma^2}\right) \text{ et } \mu_{F_{x_2}}(x_2) = \exp\left(\frac{-x_2^2}{\sigma^2}\right) \quad 1.4$$

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 1.4).

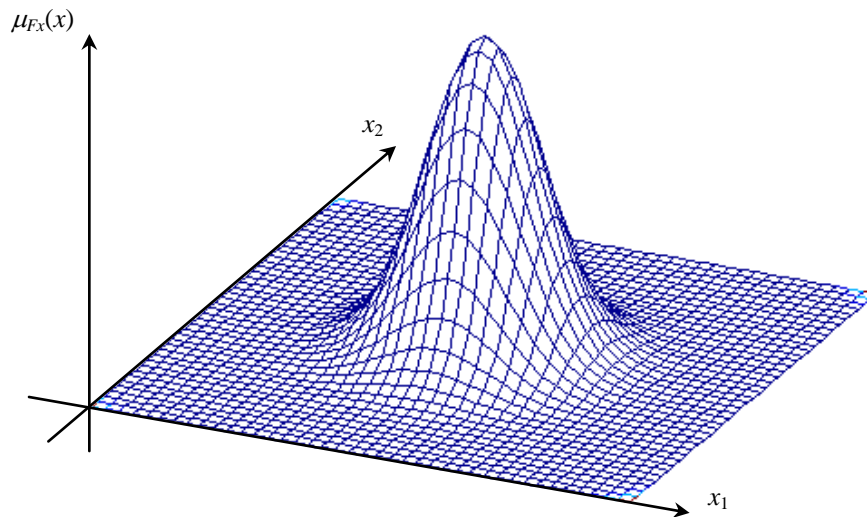


Figure 1.4 : Exemple de fonction d'appartenance gaussienne à deux dimensions

1.3.1. Partitionnement flou de l'espace d'entrée

Concernant l'antécédent du modèle flou, trois types de partitionnement peuvent être considérés [16,34 et 35] :

1.3.1.1. Partitionnement homogène

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

$$\mathfrak{R}_i : \mathbf{IF} \ x_1 \text{ is } F_{x_1}^i \ \mathbf{AND} \dots \mathbf{AND} \ x_n \text{ is } F_{x_n}^i \ \mathbf{THEN} \dots \quad 1.5$$

On remarque que ce modèle est un cas particulier de la forme générale 1.1, 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 :

$$\beta_i = \mu_{F_x^i}(x_1, x_2, \dots, x_n) = \mu_{F_{x_1}^i}(x_1) \wedge \mu_{F_{x_2}^i}(x_2) \wedge \dots \wedge \mu_{F_{x_n}^i}(x_n) \quad 1.6$$

$\beta_i (i:1,\dots,K)$: 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 1.5.

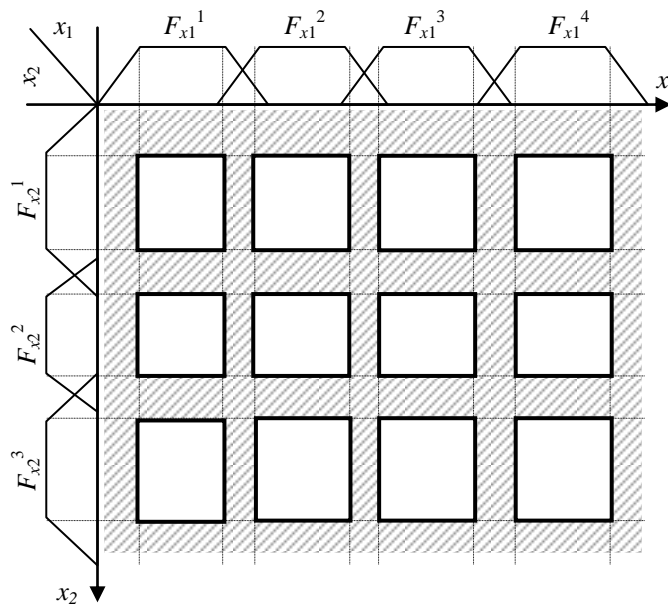


Figure 1.5 : Partition floue homogène de l'espace d'entrée schématisée par les noyaux des ensembles flous

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

$$K = N \prod_{i=1}^n M_i \quad 1.7$$

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)

1.3.1.2. 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 $F_{x_1}^1 \times F_{x_2}^3$.

La règle sera alors décrite par :

$$\mathbf{IF} \ x_1 \ \mathbf{is} \ F_{x_1}^1 \ \mathbf{AND} \ x_2 \ \mathbf{is} \ \mathbf{not} \ F_{x_2}^3 \ \mathbf{THEN} \ \dots \quad 1.8$$

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

$$\mu_{F_x^{(1)}}(x) = \mu_{F_{x_1}^1}(x_1) \wedge (1 - \mu_{F_{x_2}^3}(x_2)) \quad 1.9$$

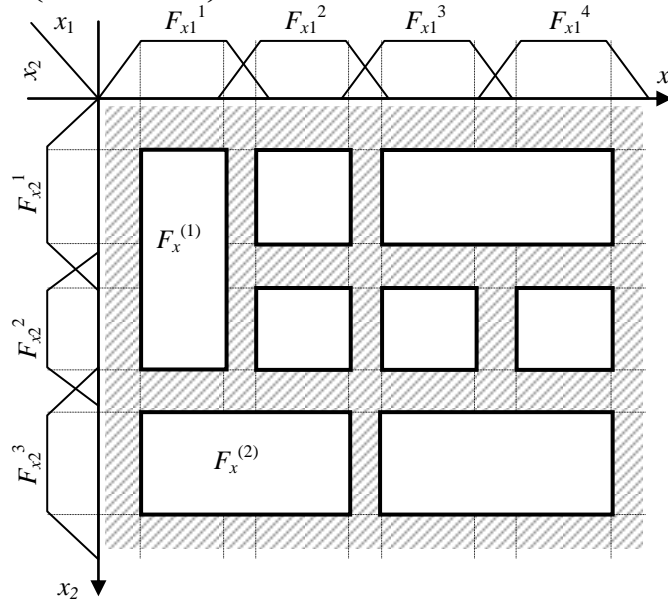


Figure 1.6: Partition floue obtenue par concaténation d'ensembles flous d'une partition homogène

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.

1.3.1.3. Partitionnement optimal

Dans le cas le plus général, il n'y a pas de restriction sur la forme des régions floues (Figure 1.7), lesquelles sont définies par des fonctions d'appartenances multivariées.

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 multivariées, 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 .

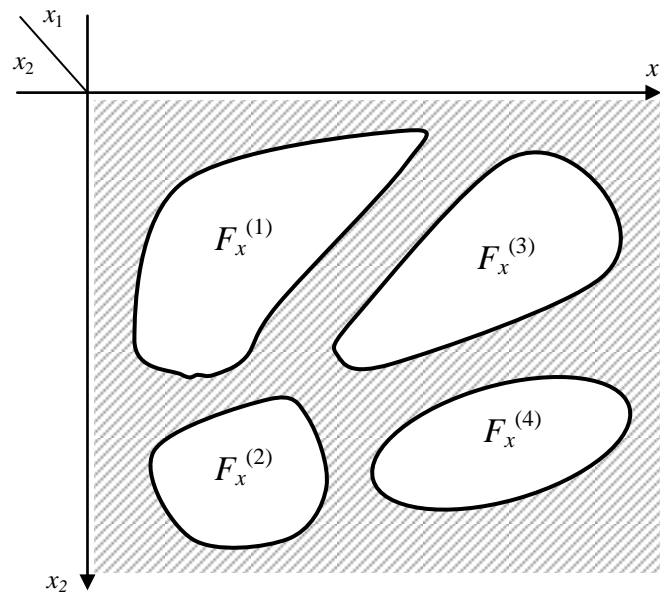


Figure 1.7 : Partition floue optimale

1.3.2. 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.

1.3.2.1. Le modèle flou linguistique

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

$$\mathfrak{R}_i : \mathbf{IF} \tilde{x} \text{ is } F_x^i \mathbf{ THEN } y \text{ is } F_y^i \quad 1.10$$

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 :

$$\begin{aligned} \mu_{F_x^i}(x) : x \in \mathbb{R}^n &\rightarrow [0,1] \\ \mu_{F_y^i}(y) : y \in \mathbb{R}^q &\rightarrow [0,1] \end{aligned} \quad 1.11$$

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 ($\tilde{x} \text{ is } F_x^i$) de la $i^{\text{ème}}$ règle et il est calculé comme suit (Figure 1.8) :

$$\beta_i = \max_{U_x} \left(\mu_A(x) \wedge \mu_{F_x^i}(x) \right) \quad ; \quad i = 1, \dots, K \quad 1.12$$

où 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 .

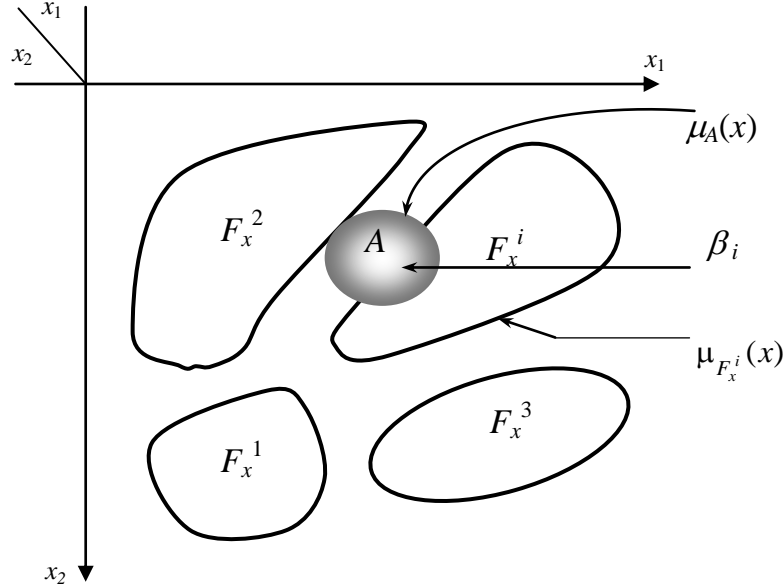


Figure 1.8 : Fuzzification ou obtention du degré de validité de l'antécédent d'une règle

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 1.5), l'équation (1.12) peut être développée comme suit :

$$\mu_A(x) \wedge \mu_{F_x^i}(x) = \left(\mu_{A^1}(x) \wedge \dots \wedge \mu_{A^n}(x) \right) \wedge \left(\mu_{F_{x_1}^i}(x) \wedge \dots \wedge \mu_{F_{x_n}^i}(x) \right) \quad 1.13$$

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

$$\begin{aligned} \mu_A(x) \wedge \mu_{F_x^i}(x) &= \left(\mu_{A^1}(x) \wedge \mu_{F_{x_1}^i}(x) \right) \wedge \dots \wedge \left(\mu_{A^n}(x) \wedge \mu_{F_{x_n}^i}(x) \right) \\ &= \bigwedge_{j=1:n} \left(\mu_{A^j}(x) \wedge \mu_{F_{x_j}^i}(x) \right) \end{aligned} \quad 1.14$$

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 :

$$\beta_i = \max_X \bigwedge_{j=1:n} \left(\mu_{A_j}(x) \wedge \mu_{F_{x_j}^i}(x) \right) = \bigwedge_{j=1:n} \max_X \left(\mu_{A_j}(x) \wedge \mu_{F_{x_j}^i}(x) \right) \quad 1.15$$

Ou encore :

$$\beta_i = \bigwedge_{j=1:n} \beta_i^j \quad 1.16$$

avec $\beta_i^j = \max_X \left(\mu_{A_j}(x) \wedge \mu_{F_{x_j}^i}(x) \right)$ comme illustré par la figure 1.9.

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 [14,15,36 et 37]. 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 [16] :

$$\mu_{B_i}(y) = \beta_i \wedge \mu_{F_y^i}(y) \quad 1.17$$

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 :

$$\mu_B(y) = S_{1 \leq i \leq K} \left(\mu_{B_i}(y) \right); \quad y \in Y \quad 1.18$$

$S(\cdot)$ 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 [16,24,34 et 38].

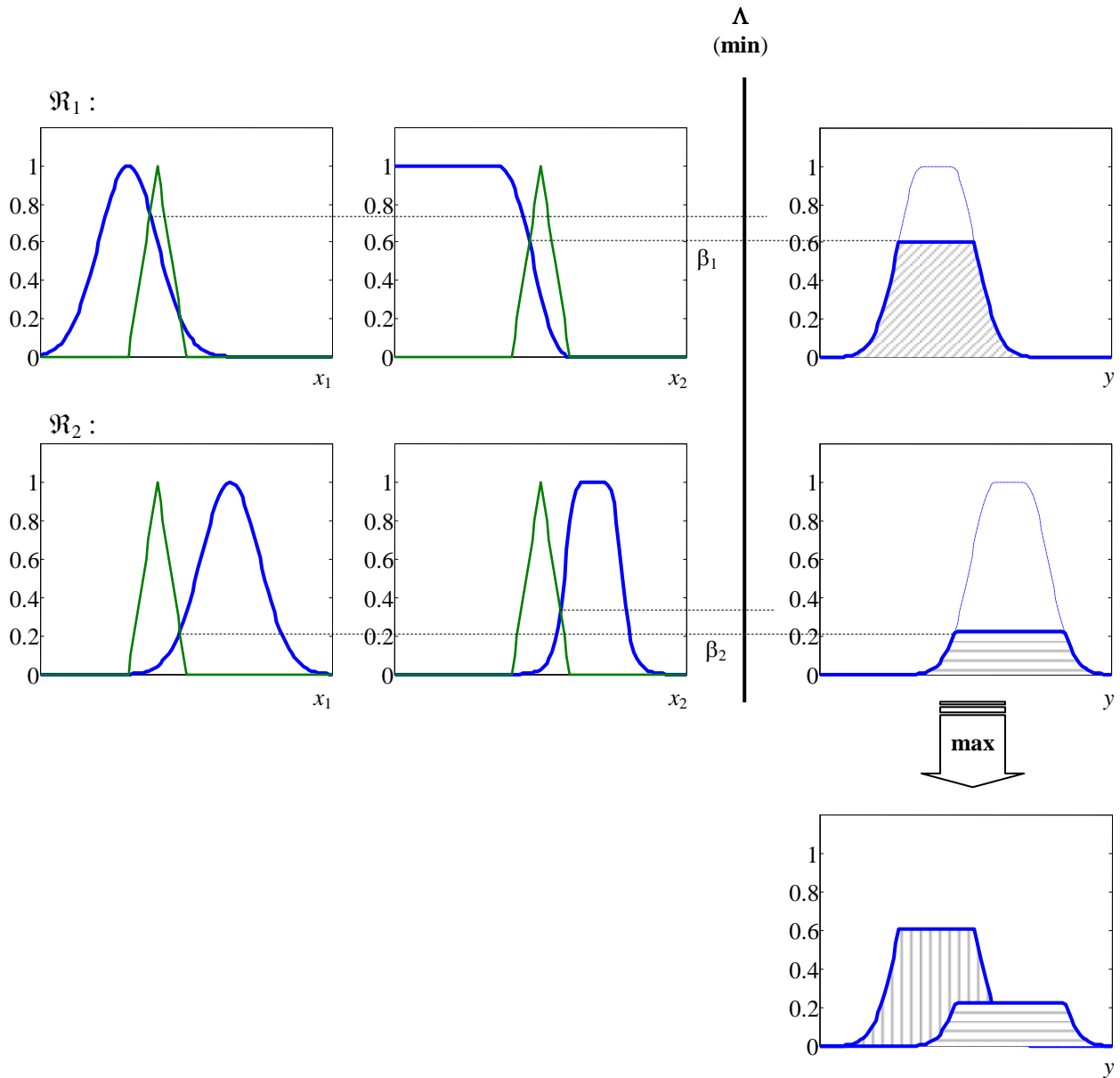


Figure 1.9 : *Système d'inférence flou de Mamdani utilisant min et max pour les opérations ET et OU respectivement*

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 [16,35,39 et 40].

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

$$\mathfrak{R}^i : \text{IF } \tilde{x} \text{ is } F_x^i \text{ THEN } y \text{ is } F_y^i \quad 1.19$$

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

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 .

$$\mathcal{R} : F_x^i \times F_y^i \rightarrow \{0, 1\} \quad 1.20$$

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 1.10.

Un modèle flou relationnel est obtenu par l'extension du modèle linguistique décrit par l'expression 1.20 :

$$\mathcal{R} : F_x^i \times F_y^i \rightarrow [0, 1] \quad 1.21$$

Dans ce cas, chaque règle contient tous les termes de la sortie. Il est attribué à chaque terme un facteur de pondération, Figure 1.11.

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 [16 et 35].

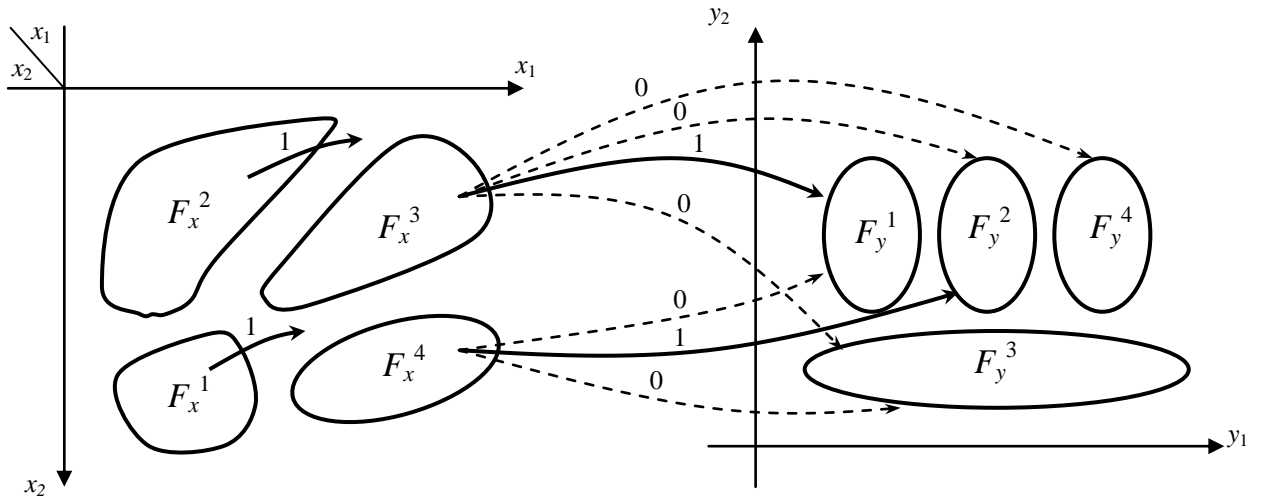


Figure 1.10 : *Modèle flou linguistique (à deux entrées et deux sorties)*

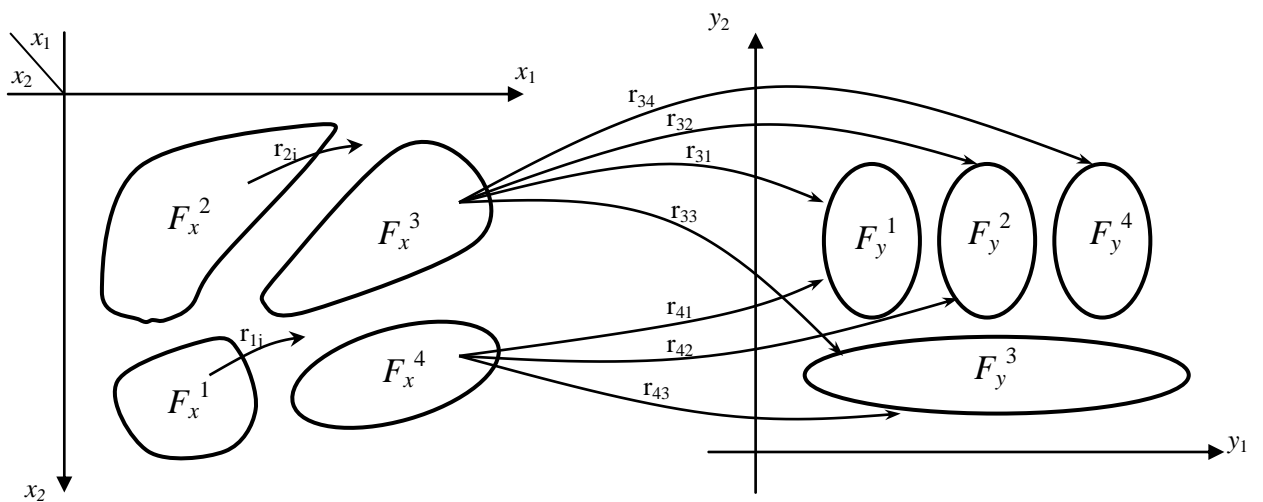


Figure 1.11 : *Modèle flou relationnel (à deux entrées et deux sorties)*

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 :

$$\beta_i = \max_x (\mu_A(x) \wedge \mu_{F_x^i}(x)) ; i = 1, \dots, K \tag{1.22}$$

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

$$\omega_j = S_{1 \leq i \leq K} (\beta_i \wedge r_{ij}) ; j = 1, \dots, M \quad 1.23$$

\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 :

$$B_j = S(\omega_j, F_y^j) \quad 1.24$$

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 [40], et défini par :

$$y_{moy} = \frac{\sum_{j=1}^M \omega_j \cdot b_j}{\sum_{j=1}^M \omega_j} \quad 1.25$$

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”.

1.3.2.2. 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 [40] :

$$\mathfrak{R}^i: \text{IF } x \text{ is } F_x^i \text{ THEN } y_i = f_i(x, a^i) ; i = 1, \dots, K \quad 1.26$$

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 [16 et 40] :

$$\mathfrak{R}^i: \text{IF } x \text{ is } F_x^i \text{ THEN } y_i = a_i^t x + b_i ; i = 1, \dots, K \quad 1.27$$

a_i : vecteur des paramètres.

b_i : valeur scalaire.

Ce modèle est appelé : *modèle linéaire de Takagi-Sugeno* (Figure 1.12).

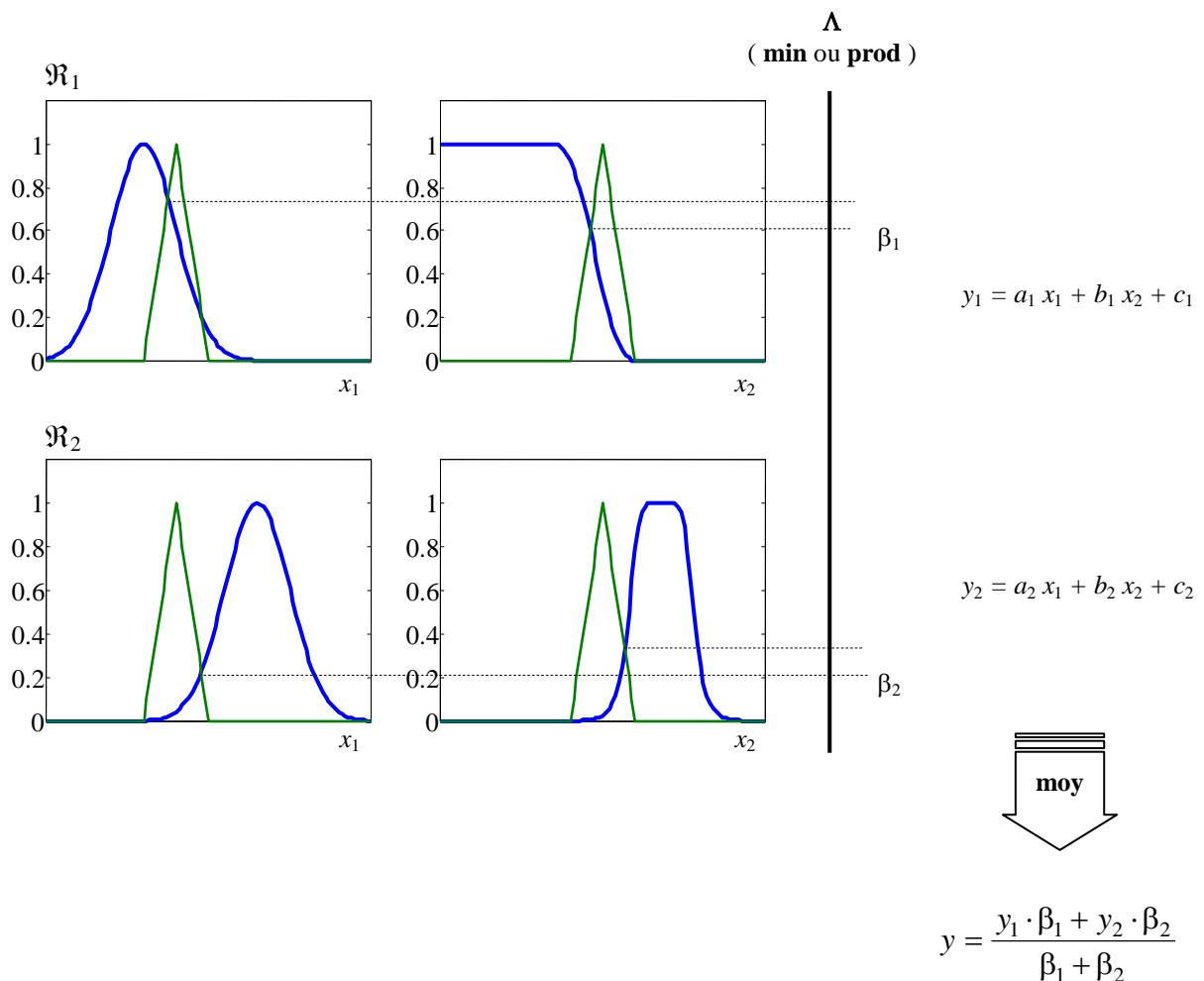


Figure 1.12 : *Système d'inférence d'un modèle linéaire flou de Takagi-Sugeno*

Mécanisme d'inférence

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

$$\beta_i(x = x_0) = \mu_{F_x^i}(x = x_0) \quad 1.28$$

2. Calcul de la sortie :

$$y_{\text{moy}} = \frac{\sum_{j=1}^K \beta_j \cdot y_j}{\sum_{j=1}^K \beta_j} = \frac{\sum_{j=1}^K \beta_j \cdot (a_i^t x + b_i)}{\sum_{j=1}^K \beta_j} \quad 1.29$$

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

$$\gamma_i = \frac{\beta_j}{\sum_{j=1}^K \beta_j} \quad ; \quad \sum_{j=1}^K \gamma_i = 1 \quad 1.30$$

la sortie du modèle sera décrite par :

$$y_{\text{moy}} = \sum_{j=1}^K \gamma_j \cdot y_j = \sum_{j=1}^K \gamma_j \cdot (a_i^t x + b_i) \quad 1.31$$

D'une autre manière, nous pouvons écrire :

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

ou bien :

$$y_{\text{moy}} = a^t(x) \cdot x + b_i \quad 1.32$$

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 :

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

$a(x)$ 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 [16,13,40 et 41].

1.3.2.3. Modèle flou de Tsukamoto

Dans les modèles flous de Tsukamoto [40], la conséquence de chaque règle, et pour chaque sortie, est représentée par un ensemble flou ayant une fonction d'appartenance monotone (Figure 1.13).

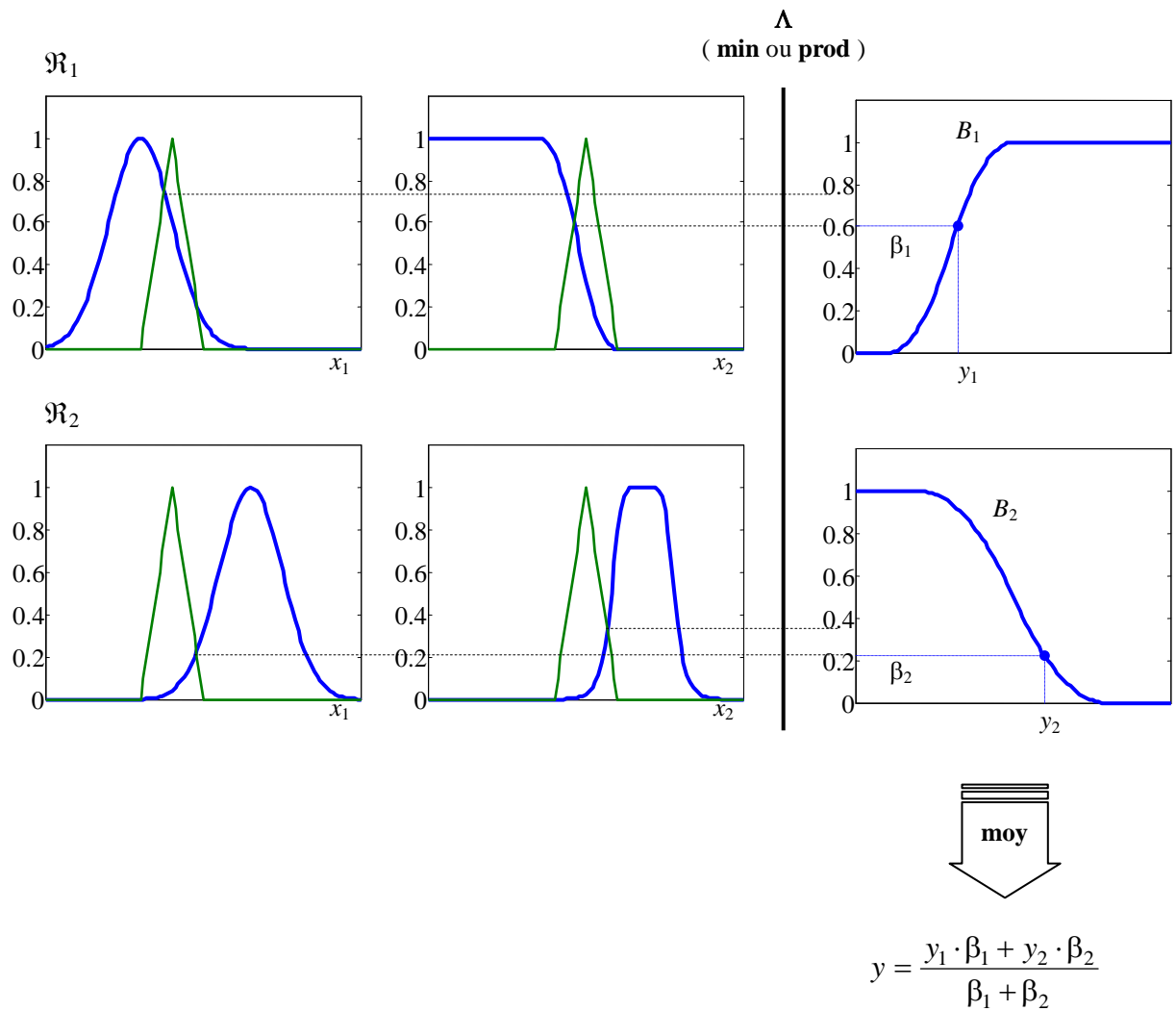


Figure 1.13 : *Système d'inférence d'un modèle flou de Tsukamoto*

1.4. Élaboration d'un modèle flou

Pour élaborer un modèle flou, deux étapes d'identification sont nécessaires : une identification structurelle et une identification paramétrique [1,2,3,16 et 28]. 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 [1 et 16].

- *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* [1,2,28 et 42].

- *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 [16].

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.

1.5. Modélisation floue linguistique des systèmes incertains

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

$$\dot{x} = f(x, \eta) + g(x, \eta) \cdot u \quad 1.33$$

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 structurelles. La figure 1.14 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.

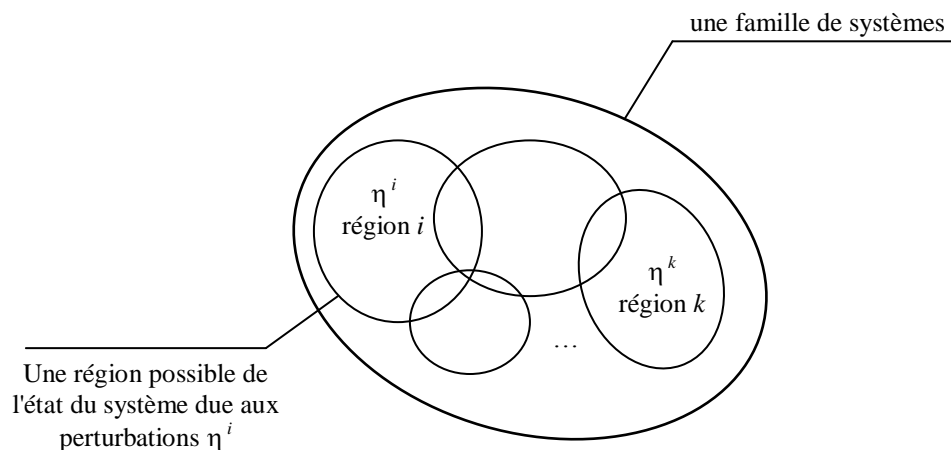


Figure 1.14 : *Effet des incertitudes sur un système*

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-Ghalia [14,15,19 et 27] consiste à produire une représentation basée sur des ensembles flous :

$$\tilde{x} = \tilde{f}(\tilde{x}) \tilde{+} \tilde{g}(\tilde{x}) \tilde{\cdot} \tilde{u} \quad 1.34$$

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

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

$$\tilde{f}(x) = \begin{pmatrix} \tilde{f}_1(\tilde{x}_1, \dots, \tilde{x}_n) \\ \vdots \\ \tilde{f}_n(\tilde{x}_1, \dots, \tilde{x}_n) \end{pmatrix}; \quad \tilde{g}(\tilde{x}) = [g_{ij}(\tilde{x}_1, \dots, \tilde{x}_n)]_{\substack{i=1..n \\ j=1..m}} \quad 1.35$$

$\tilde{+}, \tilde{\cdot}$ sont respectivement l'addition et la multiplication étendue sur les variables floues [8,15,18,19,20 et 21].

L'équation 1.34 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.

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

Concernant l'équation 1.33, 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 :

$$U_x = U_{x_1} \times U_{x_2} \times \dots \times U_{x_n} \subset \mathbb{R}^n. \quad 1.36$$

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, $\forall \eta \in P$.

Les deux dernières suppositions assurent l'existence d'une possibilité de stabilité globale du système (1.33) soumis au contrôleur approprié. Ce dernier devient lui-même Carathéodory [4,19 et 44].

1.5.2. Algorithme proposé de la modélisation floue

Pour définir les ensembles et les règles appropriés à l'équation différentielle floue (1.34), nous présentons l'algorithme constitué des étapes suivantes [18,19 et 20].

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 :

$$U_{x_i} \subseteq \bigcup_{k_i=K_{i1}}^{K_{i2}} I_{x_i}^{k_i} ; i = 1, \dots, n \quad 1.37$$

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

$$P_F(U_{x_i}) = \left\{ F_{x_i}^{k_i} / k_i = K_{i1}, \dots, K_{i2} \right\} \quad 1.38$$

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 :

$$F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \quad 1.39$$

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

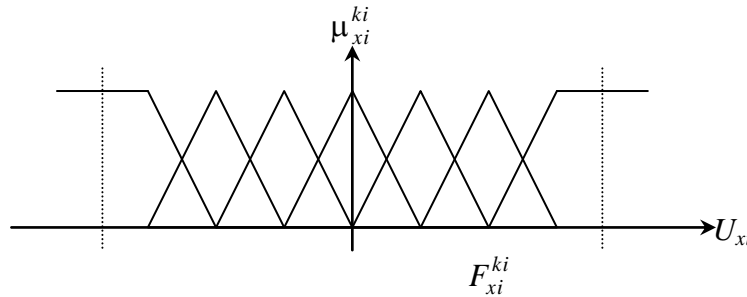


Figure 1.15 : Répartition des ensembles flous pour une variable d'entrée

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 $I_{\tilde{f}_i}^{(k_1, \dots, k_n)}$ $i = 1, \dots, m$ de la manière suivante :

$$I_{\tilde{f}_i}^{(k_1, \dots, k_n)} = \left\{ f_i(x(t), \eta(t)) \mid x_1(t) \in I_{x_1}^{k_1}, \dots, x_n(t) \in I_{x_n}^{k_n}, \eta(t) \in P \right\} \quad 1.40$$

Ou bien :

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

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 :

$$U_{f_i} = \bigcup_{(k_1, \dots, k_n)} I_{f_i}^{(k_1, \dots, k_n)} = \bigcup_{k_1=K_{11}}^{K_{12}} \bigcup_{k_2=K_{21}}^{K_{22}} \dots \bigcup_{k_n=K_{n1}}^{K_{n2}} I_{f_i}^{(k_1, \dots, k_n)} \quad 1.41$$

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 $f_i(\cdot)$:

$$I_{g_{ij}}^{(k_1, \dots, k_n)} = \left\{ g_{ij}(x(t), \eta(t)) \quad / \quad x(t) \in I_x^{(k_1, \dots, k_n)}, \quad \eta(t) \in P \right\} \quad 1.42$$

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} :

$$U_{g_{ij}} = \bigcup_{(k_1, \dots, k_n)} I_{g_{ij}}^{(k_1, \dots, k_n)} = \bigcup_{k_1=K_{11}}^{K_{12}} \bigcup_{k_2=K_{21}}^{K_{22}} \dots \bigcup_{k_n=K_{n1}}^{K_{n2}} I_{g_{ij}}^{(k_1, \dots, k_n)} \quad 1.43$$

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 bornées 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 1.16).

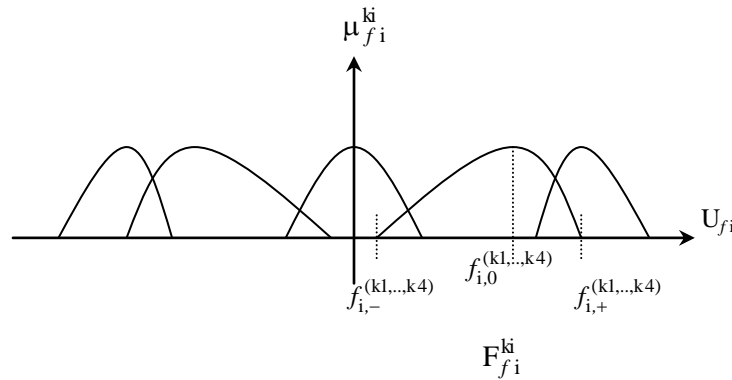


Figure 1.16 : Répartition des ensembles flous pour une fonction de sortie

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

$$P_F(U_{f_i}) = \left\{ F_{f_i}^{(k_1, \dots, k_n)} \quad / \quad k_\ell = K_{\ell_1}, \dots, K_{\ell_2} \quad ; \quad \ell = 1, \dots, n \right\} \quad 1.44$$

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 $U_{g_{ij}}$:

$$P_F(U_{g_{ij}}) = \left\{ F_{g_{ij}}^{(k_1, \dots, k_n)} \quad / \quad k_\ell = K_{\ell_1}, \dots, K_{\ell_2} \quad ; \quad \ell = 1, \dots, n \right\} \quad 1.45$$

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

Etape 4 : Dédution 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 :

$$\begin{aligned} \tilde{f}(\cdot) : P_F(U_{x1}) \times \dots \times P_F(U_{xn}) &\rightarrow [P_F(\mathfrak{R})]^n \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{f}(\tilde{x}) \end{aligned} \quad 1.46$$

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 :

$$\begin{aligned} \tilde{g}(\cdot) : P_F(U_{x1}) \times \dots \times P_F(U_{xn}) &\rightarrow [P_F(\mathfrak{R})]^{n \times m} \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{g}(\tilde{x}) \end{aligned} \quad 1.47$$

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 :

$$\begin{aligned} \mathfrak{R}_{f_i}^{(k_1, \dots, k_n)} : \text{IF } x_1 \text{ is } F_{x_1}^{k_1} \text{ AND } \dots \text{ AND } x_n \text{ is } F_{x_n}^{k_n} \\ \text{THEN } f_i \text{ is } F_{f_i}^{(k_1, \dots, k_n)} \end{aligned} \quad 1.48$$

$$\begin{aligned} \text{ou } \mathfrak{R}_{f_i}^{(k_1, \dots, k_n)} : \text{ IF } \quad x \text{ is } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \\ \text{THEN } f_i \text{ is } F_{f_i}^{(k_1, \dots, k_n)} \end{aligned} \quad 1.49$$

Et de même :

$$\begin{aligned} \mathfrak{R}_{g_{ij}}^{(k_1, \dots, k_n)} : \text{ IF } \quad x_1 \text{ is } F_{x_1}^{k_1} \text{ AND } \dots \text{ AND } x_n \text{ is } F_{x_n}^{k_n} \\ \text{THEN } g_{ij} \text{ is } F_{g_{ij}}^{(k_1, \dots, k_n)} \end{aligned} \quad 1.50$$

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

$$\begin{aligned} \mathfrak{R}_f^{(k_1, \dots, k_n)} : \text{ IF } \quad x \text{ est } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \\ \text{THEN } f(x, \eta) \text{ est } F_f^{(k_1, \dots, k_n)} = [F_{f_1}^{(k_1, \dots, k_n)}, F_{f_2}^{(k_1, \dots, k_n)}, \dots, F_{f_n}^{(k_1, \dots, k_n)}]^t \end{aligned} \quad 1.51$$

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

De même:

$$\begin{aligned} \mathfrak{R}_g^{(k_1, \dots, k_n)} : \text{ IF } \quad x \text{ est } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \\ \text{THEN } g(x, \eta) \text{ est } F_g^{(k_1, \dots, k_n)} = [F_{g_{ij}}^{(k_1, \dots, k_n)}]_{\substack{i=1, \dots, n \\ j=1, \dots, m}} \end{aligned} \quad 1.52$$

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.

1.5.3. 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 1.16), on peut générer l'entrée floue \tilde{u} et l'écrire sous forme d'un vecteur de fonctions floues :

$$\begin{aligned} \tilde{u}(\cdot) : P_F(U_{x1}) \times \dots \times P_F(U_{xn}) &\rightarrow [P_F(\mathfrak{R})]^m \\ \tilde{x} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n}) &\rightarrow \tilde{u}(\tilde{x}) \end{aligned} \quad 1.53$$

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 :

$$\begin{aligned} \mathfrak{R}_u^{(k_1, \dots, k_n)} : \text{ IF } \quad x \text{ is } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \\ \text{ THEN } \quad u \text{ is } F_u^{(k_1, \dots, k_n)} = [F_{u_1}^{(k_1, \dots, k_n)}, F_{u_2}^{(k_1, \dots, k_n)}, \dots, F_{u_n}^{(k_1, \dots, k_n)}]^t \end{aligned} \quad 1.54$$

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

1.5.4. Obtention du modèle flou

A partir des équations (1.34), (1.51), (1.52) et (1.54), le modèle flou peut être représenté par les $\prod_{i=1}^n M_i$ règles suivantes :

$$\begin{aligned} \mathfrak{R}_{\dot{x}}^{(k_1, \dots, k_n)} : \text{ IF } \quad x \text{ is } F_x^{(k_1, \dots, k_n)} = [F_{x_1}^{k_1}, F_{x_2}^{k_2}, \dots, F_{x_n}^{k_n}]^t \\ \text{ THEN } \quad \dot{x} \text{ is } F_{\dot{x}}^{(k_1, \dots, k_n)} = [F_{\dot{x}_1}^{(k_1, \dots, k_n)}, F_{\dot{x}_2}^{(k_1, \dots, k_n)}, \dots, F_{\dot{x}_n}^{(k_1, \dots, k_n)}]^t \end{aligned} \quad 1.55$$

avec :

$$F_{\dot{x}_i}^{(k_1, \dots, k_n)} = F_{f_i}^{(k_1, \dots, k_n)} \tilde{\ddagger} \sum_{j=1}^m F_{g_{ij}}^{(k_1, \dots, k_n)} \tilde{\sim} F_{u_j}^{(k_1, \dots, k_n)} \quad 1.56$$

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

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

1.6. 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 [8,16,17,19,29 et 20], comme dans le cas des machines électriques.

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 1.17).

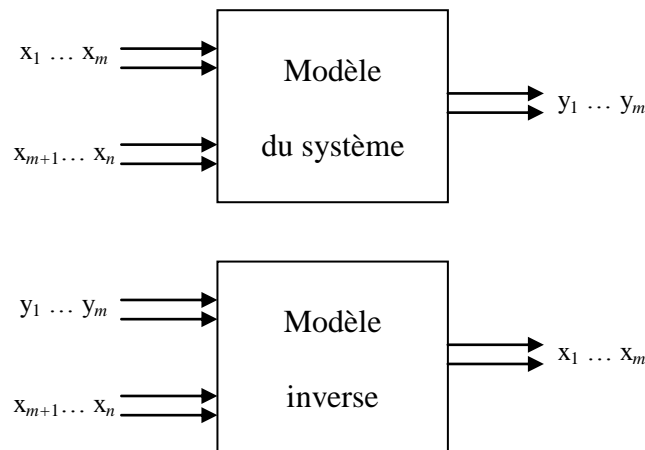


Figure 1.17 : *Le modèle inverse d’un système MIMO*

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 [8,19 et 20], en modélisant, d’abord, le système en introduisant la logique floue et en inversant ensuite la base de règles appropriée [19 et 20],
- directement à partir d’une représentation analytique du modèle inverse, tout en tenant compte des incertitudes du système [8 et 19].

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.

1.6.1. 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 1.18. Néanmoins, il est important de remarquer que la table de règles qui en résulte se trouve être incomplète, ce qui représente un inconvénient majeur.

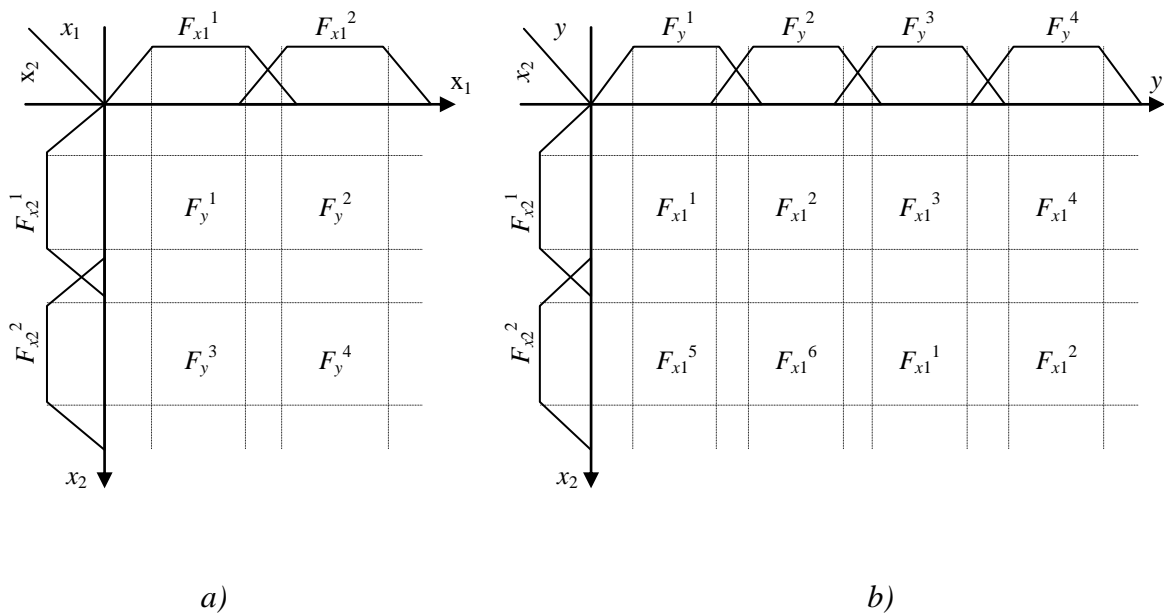


Figure 1.18 : Exemple d'inversion d'une base de règles
 a) modèle flou original ; b) modèle inverse flou

Afin de pallier à ce problème, nous utilisons une des techniques présentée par Babuska [16] qui consiste à représenter au préalable géométriquement le modèle flou dans un espace

à trois dimensions comme le montre la figure 1.19. 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 1.20). 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 (1.1a) et (1.1b) 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 1.1 : 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.

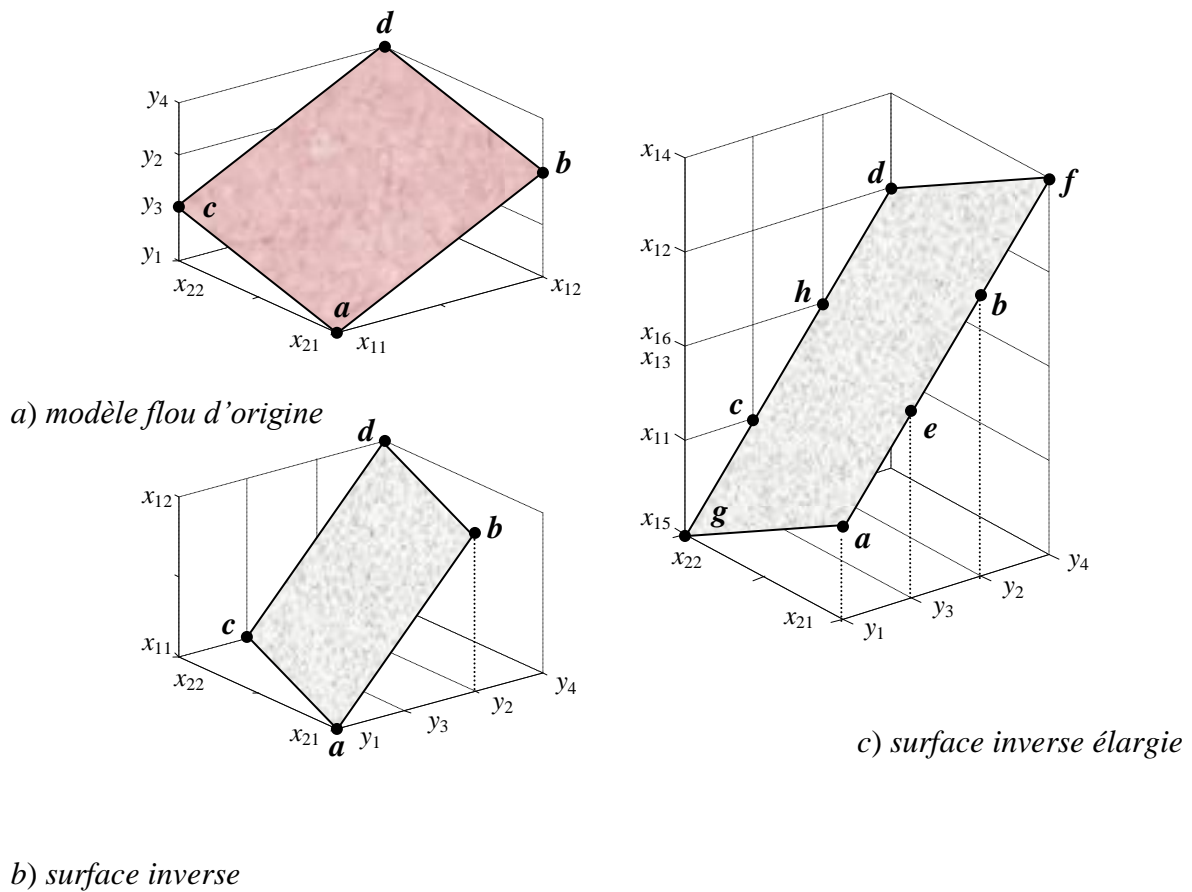


Figure 1.21 : Inversion d'une base de règle de deux entrées et une sortie

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 1.21c). 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.

1.6.2. 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 :

$$y = f(x) \tag{1.57}$$

$y = (y_1, \dots, y_m)^t$: 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 (1.57) devient :

$$y = f(x, \eta) \tag{1.58}$$

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 :

$$\tilde{y} = \begin{pmatrix} \tilde{y}_1 \\ \dots \\ \tilde{y}_m \end{pmatrix} = \tilde{f}(\tilde{x}) = \begin{pmatrix} \tilde{f}_1(\tilde{x}_1, \dots, \tilde{x}_n) \\ \dots \\ \tilde{f}_m(\tilde{x}_1, \dots, \tilde{x}_n) \end{pmatrix} \tag{1.59}$$

\tilde{x} et \tilde{y} : vecteurs flous correspondant aux vecteurs x et y .

\tilde{f} : vecteur de fonctions floues.

La technique de modélisation floue proposée par Ben-Ghali [8,19 et 20], et que nous avons adoptée, permet de concevoir un modèle flou robuste pour les systèmes incertains, sans avoir à connaître la structure ou la forme des incertitudes.

1.7. 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 2

Chapitre 2

Application du MIF à la Commande de la MAS

2.1 Introduction

On ne présente plus la machine asynchrone, c'est le moteur utilisé au quotidien dans les applications domestiques (machine à laver), à l'atelier, à l'usine (machinerie d'entraînement) ainsi que dans les domaines spécialisés (traction ferroviaire, maritime, auxiliaires dans les centrales...). Sa robustesse et son coût d'achat et d'entretien lui ont permis de conquérir un espace de plus en plus grand au détriments des machines synchrones et à courant continu.

De façon générale, il est incontestable que la motorisation électrique commandée prend une place de plus en plus large [4,9 et 50]. Il ne faut cependant pas non plus négliger l'immense parc de moteurs asynchrones utilisés à des fins d'entraînement non commandé mais la vitesse variable est en plein essor et tout porte à croire que la demande va s'accroître surtout avec la baisse continue des prix de l'électronique de puissance et de commande.

Nous présenterons la modélisation adoptée en vue de la commande de la machine ainsi que la commande vectorielle classique utilisée comme base de comparaison. Nous verrons alors l'utilisation des méthodes de commande basées sur la modélisation floue des systèmes incertains.

2.2 Contrôle classique de la MAS

2.2.1 Introduction

La machine asynchrone à cage dont le rotor ne tourne pas à la vitesse du champ tournant et dont la seule entrée électrique est au stator, pose des problèmes difficiles pour sa commande. La communauté scientifique et industrielle a imaginé bien des méthodes de commande afin de pouvoir la contrôler en couple, en vitesse ou en position. Les méthodes scalaires sont très prisées pour leur simplicité de mise en œuvre. Cependant, elles ne peuvent pas garantir du couple à l'arrêt ni d'obtenir la dynamique et la précision des méthodes dites vectorielles [4,9,42,49 et 50]. La commande vectorielle a comme principe de commande par modèle inverse (figure 2.1) ;

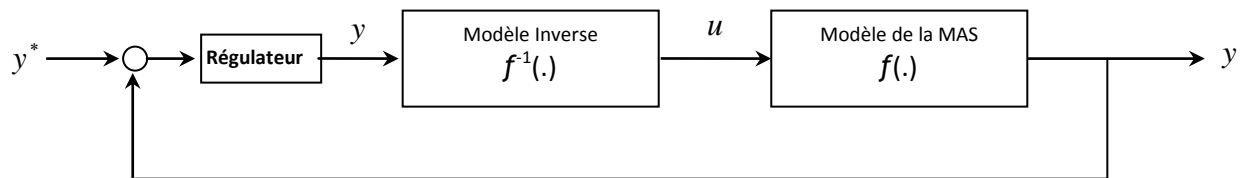


Fig.2.1. Principe de Base de la CFO

Elle est apparue avec les travaux de Blaschke [4,9,46,47,49 et 50]. Elle n'a cependant pas eu tout de suite un grand essor car les régulations, à l'époque, reposaient sur des composants analogiques, l'implantation de la commande était alors difficile.

Avec l'avènement des microcontrôleurs et des dispositifs permettant le traitement du signal DSP [26], il est devenu possible de réaliser une telle commande à un coût raisonnable. Cela a conduit à une explosion des recherches et des applications relatives à la commande vectorielle de la machine asynchrone. Le nombre des publications, des applications et des brevets en témoigne.

Si beaucoup de problèmes sont résolus, certains autres font encore l'objet de recherche. Quand on ne cherche pas à obtenir des performances élevées, même si l'identification n'est pas bien faite et que les régulateurs du schéma de contrôle vectoriel ne sont pas réglés à l'optimum, le comportement global du système commande-convertisseur-machine paraît satisfaisant.

Cependant, les problèmes ne surgissent que lorsque la machine est poussée dans ses retranchements.

2.2.2. Modèle de la machine asynchrone

Dans le cadre de ce travail, nous nous sommes intéressés aux modèles de la machine asynchrone qui permettent de simuler son fonctionnement en régimes transitoires ainsi qu'à ceux qui débouchent sur une commande suivant un schéma de contrôle vectoriel par orientation du flux rotorique.

2.2.2.1. Hypothèses de travail

Pour cette mise en équation, nous supposons que le bobinage est réparti de manière à donner une f.m.m. sinusoïdale s'il est alimenté par des courants sinusoïdaux.

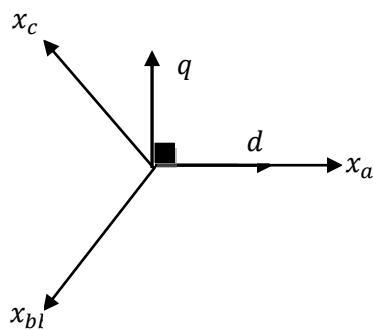
Nous supposons également que nous travaillons en régime non saturé. Nous négligeons le phénomène d'hystérésis, les courants de Foucault et l'effet de peau.

Enfin, le régime homopolaire est nul puisque le neutre n'est pas relié.

2.2.2.2. Les équations de la machine asynchrone en régime quelconque

La mise en équation de la machine asynchrone avec les hypothèses que nous avons retenues étant classique, nous ne mentionnerons que les points qui nous semblent essentiels et les choix qui nous sont propres par rapport à ce qui se fait habituellement. Pour plus de détail, le lecteur pourra se référer à [9,46,47 et 49].

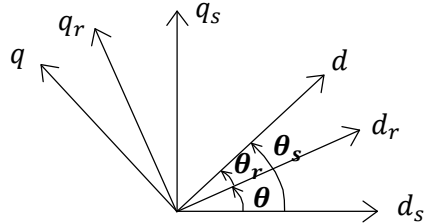
Précisons tout d'abord que nous utiliserons la transformation de Park qui est bien pratique en commande où l'on traite des grandeurs (i_{ds}, i_{qs}) .



$$\begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} = \frac{P = \sqrt{2/3}}{} \begin{bmatrix} \cos(\theta) & \cos(\theta - 2\pi/3) & \cos(\theta + 2\pi/3) \\ -\sin(\theta) & -\sin(\theta - 2\pi/3) & -\sin(\theta + 2\pi/3) \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} x_d \\ x_q \\ x_c \end{bmatrix}$$

$$\begin{bmatrix} x_d \\ x_q \end{bmatrix} = \frac{P^{-1} = \sqrt{2/3}}{\phantom{P^{-1}}} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 1/\sqrt{2} \\ \cos(\theta - 2\pi/3) & -\sin(\theta - 2\pi/3) & 1/\sqrt{2} \\ \cos(\theta + 2\pi/3) & -\sin(\theta + 2\pi/3) & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix}$$

Il apparaît clairement ensuite que les repères de la transformation de Park des grandeurs statoriques et celle des grandeurs rotoriques doivent coïncider pour simplifier les équations. Ceci se fait en liant les angles θ^s et θ^r par la relation :



$$\theta^s = \theta + \theta^r \quad 2.1$$

Les flux dans ce système d'axes s'écrivent :

$$\begin{cases} \Phi_{ds} = L_s i_{ds} + L_M i_{dr} \\ \Phi_{qs} = L_s i_{qs} + L_M i_{qr} \end{cases} \quad \begin{cases} \Phi_{dr} = L_r i_{dr} + L_M i_{ds} \\ \Phi_{qr} = L_r i_{qr} + L_M i_{qs} \end{cases} \quad 2.2$$

et le couple électromagnétique :

$$C_{em} = \frac{pL_M}{L_r} [\Phi_{dr} i_{qs} - \Phi_{qr} i_{ds}] \quad 2.3$$

Notons $w_s = \dot{\theta}_s$ que nous appellerons pulsation statorique, bien que le régime puisse être quelconque (transitoire non sinusoïdal). De même, nous noterons $w_r = \dot{\theta}_r$ la pulsation rotorique et $w = w_s - w_r = \dot{\theta} = p\Omega$ la pulsation mécanique.

Nous pouvons alors écrire :

$$\begin{cases} V_{ds} = R_s i_{ds} + \frac{d\Phi_{ds}}{dt} - \omega_s \Phi_{qs} \\ V_{qs} = R_s i_{qs} + \frac{d\Phi_{qs}}{dt} + \omega_s \Phi_{ds} \\ 0 = R_r i_{dr} + \frac{d\Phi_{dr}}{dt} - (\omega_c - \omega_r) \Phi_{qr} \\ 0 = R_r i_{qr} + \frac{d\Phi_{qr}}{dt} + (\omega_c - \omega_r) \Phi_{dr} \end{cases} \quad 2.4$$

En ajoutant l'équation mécanique : $J \frac{d\Omega_r}{dt} = C_{em} - C_r - f\Omega_r$ (2.5) et en réarrangeant les équations de façon à faire apparaître comme vecteur d'état $[i_{ds} \ i_{qs} \ \Phi_{dr} \ \Phi_{qr} \ \Omega]$, on obtient :

$$\left\{ \begin{array}{l} \frac{d\Omega_r}{dt} = \frac{p_m L_m}{J L_r} (\phi_{dr} i_{qs} - \phi_{qr} i_{ds}) - \frac{f_v}{J} \Omega_r - \frac{T_r}{J} \\ \frac{d\phi_{dr}}{dt} = \frac{R_r}{L_r} L_m i_{ds} - \frac{R_r}{L_r} \phi_{dr} + p \Omega_r \phi_{qr} \\ \frac{d\phi_{qr}}{dt} = \frac{R_r}{L_r} L_m i_{qs} - \frac{R_r}{L_r} \phi_{qr} - p \Omega_r \phi_{dr} \\ \frac{di_{ds}}{dt} = -\frac{L_m R_r}{\sigma L_s L_r^2} \phi_{dr} - \frac{p L_m}{\sigma L_s L_r} \Omega_r \phi_{qr} - \frac{L_m^2 R_r + L_r^2 R_{s1}}{\sigma L_s L_r^2} i_{ds} + \omega_s i_{qs} + \frac{V_{ds}}{\sigma L_s} \\ \frac{di_{qs}}{dt} = \frac{M_q R_r}{\sigma L_s L_r^2} \phi_{qr} - \frac{p L_m}{\sigma L_s L_r} \Omega_r \phi_{dr} - \frac{L_m^2 R_r + L_r^2 R_{s2}}{\sigma L_s L_r^2} i_{qs} - \omega_s i_{ds} + \frac{V_{qs}}{\sigma L_s} \end{array} \right. \quad 2.6$$

En effet seules les valeurs instantanées des tensions et de la pulsation statorique, et du couple résistant doivent être déterminées pour les imposer à la machine.

Le but de la commande vectorielle est d'arriver à commander la machine asynchrone comme une machine à courant continu à excitation indépendante où il y a un découplage naturel entre la grandeur commandant le flux, le courant d'excitation, et celle liée au couple, le courant d'induit. Ce découplage permet d'obtenir une réponse très rapide du couple.

En parlant d'orientation du flux, c'est plutôt le système d'axe d-q que l'on oriente de manière à ce que l'axe d soit en phase avec le flux, c'est à dire [43,44 et 47]:

$$\Phi_{dr} = \Phi_r ; \quad \Phi_{qr} = 0 \quad 2.7$$

La commande vectorielle à orientation du flux rotorique est la plus utilisée car elle élimine l'influence des réactances de fuite rotorique et statorique et donnent de meilleurs résultats que les méthodes basées sur l'orientation du flux statorique ou d'entrefer [4,9 et 44]. En imposant $\Phi_{qr} = 0$, les équations de la machine dans un référentiel lié au champ tournant deviennent :

$$\begin{cases}
 v_{ds} = \frac{R_s}{L_m} \Phi_r - \frac{R_r(L_s L_r - L_m^2)}{L_m p^2} C_e^2 - \frac{(L_s L_r - L_m^2)}{L_m} \Omega C_e \\
 v_{qs} = \frac{p L_s}{L_m} \Omega \Phi_r + \frac{(L_r R_s + L_s R_r)}{p L_m} \frac{C_e}{\Phi_r} + \frac{(L_s L_r - L_m^2)}{p L_m} \frac{1}{\Phi_r} \frac{dC_e}{dt} \\
 \omega_s = \frac{R_r}{p} \frac{C_e}{\Phi_r^2} + p \Omega \\
 \Phi_{dr} = \Phi_r = \frac{L_m / T_r}{s + 1/T_r} \quad , \quad i_{ds} = \frac{\Phi_r}{L_m} \\
 C_e = \frac{p L_m}{L_r} \Phi_r i_{qs}
 \end{cases} \quad 2.8$$

Dans cette commande, l'angle de Park θ^s est calculé à partir de la pulsation statorique, elle-même reconstituée à l'aide de la vitesse de la machine et de la pulsation rotorique θ^r . La commande vectorielle est dite à boucle ouverte s'il n'y a pas de régulation de flux [2 et 4]. Le flux est imposé dans ce cas par i_{ds} , de plus la pulsation statorique peut uniquement être estimée par la relation (2.9). Dans ce type de commande, l'angle θ_s utilisé pour la transformation directe et inverse est calculé à partir de la formule suivante :

$$\theta_s = \int \left(p \Omega + \frac{i_{qs}}{\tau_r i_{ds}} \right) dt \quad \text{ou} \quad i_{ds} = \frac{\Phi_r}{L_m} \quad 2.9$$

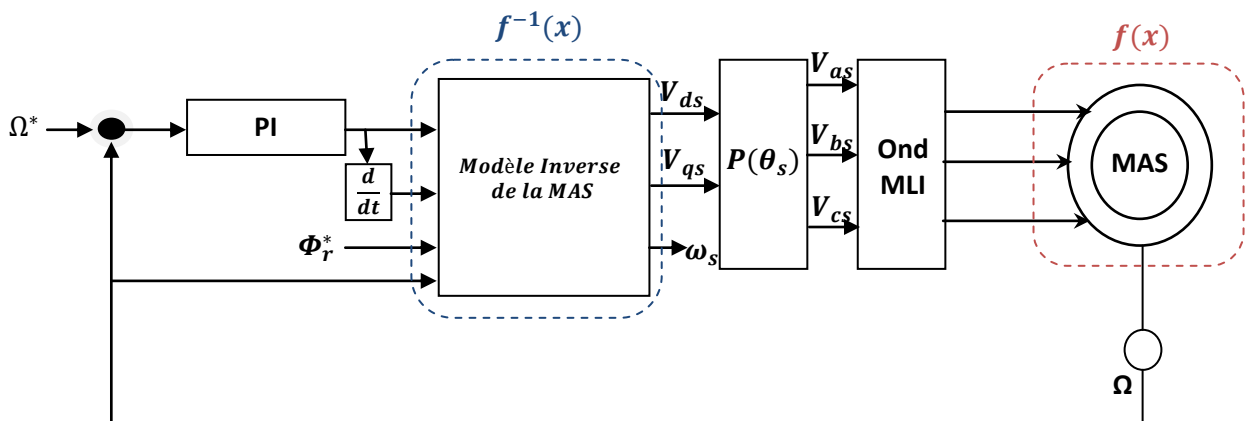


Figure 2.2 Régulation de vitesse par l'approche classique

La figure 2.2 représente le schéma bloc d'une régulation de vitesse du moteur asynchrone commandé par orientation du flux rotorique (commande par modèle inverse). Les principaux constituants dans ce type de commande sont la boucle de régulation de vitesse, et le modèle inverse de la machine qui donne les tensions d'alimentations et la pulsation statorique.

La vitesse est régulée à travers la boucle de vitesse. La sortie de son régulateur est le Couple électromagnétique de C_e^* référence, il est limité de manière à tenir compte des caractéristiques des IGBT de l'onduleur [48 et 64] et de la surcharge de la machine. Le courant i_{ds} est calculé à partir du flux à imposer. Ce flux correspond à sa valeur nominale pour la zone de vitesse inférieure à la vitesse de base. Au delà de cette zone, on procède au "défluxage" de la machine de manière à pouvoir atteindre des vitesses supérieures. Le couple maximal que l'on peut imposer devient alors plus faible. Le procédé de défluxage en grande vitesse est utilisé en particulier en traction électrique où l'on a besoin d'un fort couple pendant la phase de démarrage et d'un couple plus faible (qui ne sert à lutter que contre les frottements) pendant la marche normale.

Le modèle inverse de la machine donne les deux tensions de références v_{ds} et v_{qs} et la pulsation synchrone ω_s . Les deux tensions sont alors transformées en grandeurs triphasées à l'aide d'une rotation d'angle θ_s . L'onduleur à MLI applique des créneaux de tensions à la machine dont les valeurs moyennes sur une période de MLI correspondent aux valeurs v_{ds} . En parallèle, la pulsation statorique puis l'angle θ_s sont calculés à partir d'une mesure de vitesse mécanique et du calcul de la pulsation de glissement. C'est cet angle qui sera utilisé dans la transformation de Park.

En analysant ce schéma de commande et les équations associées, on voit apparaître principalement deux paramètres : L_m et T_r . Ils lient le flux rotorique et le courant i_{ds} qui le contrôle, mais ils apparaissent surtout dans la formule qui permet de calculer ω_s . Une surestimation ou une sous-estimation de la constante de temps rotorique conduisent respectivement à une surexcitation ou une sous-excitation de la machine. Dans les deux cas, l'amplitude et la phase du flux rotorique ne sont pas celles que l'on voudrait imposer, il en résulte une dégradation des performances, voire une instabilité du système. On perd alors le contrôle vectoriel. Cette dépendance vis-à-vis des paramètres de la machine peut être

réduite en utilisant les techniques de la modélisation intelligente par la logique floue [43, 46 et 47].

2.2.3. Régulation, méthodes classiques

Le système étant un système discret, les coefficients du régulateur équivalent dans un système continu ne correspondent pas directement à celui qu'il faut implanter dans les programmes de régulation, que ce soit pour la simulation ou pour l'expérimentation. Une des approches pour le dimensionnement des régulateurs des systèmes échantillonnés consiste à concevoir le régulateur en considérant le système comme continu, mais en y introduisant les retards inhérents à la régulation numérique, puis à calculer le régulateur équivalent discret.

Nous représentons les retards du capteur et filtre de vitesse et le convertisseur statique respectivement par $e^{-pT_{wfc}}$ et $e^{-pT_{wo}}$. Ces retards seront approximer par des fonctions de transferts de premier ordre. Le schéma bloc de régulation de la vitesse est le suivant :

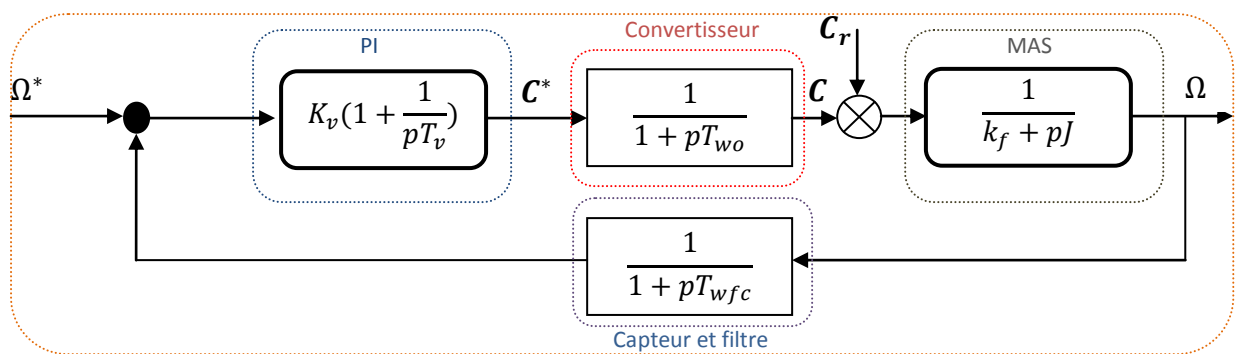


Figure 2.3 Boucle de régulation de la vitesse, structure PI

avec :

K_v, T_v : coefficients du PI.

T_{wo} : délai dans introduit par le convertisseur.

T_{wfc} : délai introduit par le capteur et filtrage de la vitesse.

La fonction de transfert en boucle ouverte par rapport à la consigne :

$$G_{ov} = K_v \left(1 + \frac{1}{pT_v}\right) \cdot \frac{1}{1+pT_{wo}} \cdot \frac{1}{k_f+pJ} \cdot \frac{1}{1+pT_{wfc}} \quad 2.10$$

Et en boucle fermée :

$$G_{fv} = (1 + pT_{wfc}) \cdot \frac{G_{ov}}{1+G_{ov}} \quad 2.11$$

Afin d'évaluer les performances de la commande, des simulations et des évaluations expérimentales ont été réalisées. Dans notre cas, nous avons tenu compte des problèmes dus à la commande numérique, en particulier, l'effet d'échantillonnage. A chaque période d'échantillonnage, la vitesse du rotor est acquise puis les tensions de références V^*_{ds} et V^*_{qs} sont obtenues à l'aide du modèle inverse du moteur en tenant compte de la condition du flux orienté. Les tensions de référence des trois phases obtenues après transformation de Park sont envoyées au générateur des signaux MLI qui fournit les séquences d'alimentation des trois phases en utilisant les nouvelles valeurs de référence. Les résultats de simulations et ceux de l'expérimentation pour une consigne de vitesse de -120 rad/sec à 140 rad/sec sont donnés sur les Figure 2.4 et 2.5 respectivement. Les courbes représentées donnent les réponses de la vitesse rotorique, couple électromagnétique, du courant statorique et de la tension statorique. On remarque que La vitesse atteint assez rapidement sa référence. Elles montrent le bon fonctionnement de cette commande dans des conditions normales de fonctionnement. La vitesse se stabilise à sa nouvelle valeur de référence. La commande permet donc d'assurer un bon fonctionnement dans les deux quadrants. L'analyse des interfaces entre commande et puissance nous a permis de faire une synthèse de régulateur de vitesse en prenant en compte les divers retards du système ainsi que le phénomène d'échantillonnage. Il est intéressant de remarquer que les résultats de simulation concordent avec ceux issus des essais expérimentaux lors de validation de l'algorithme de commande. Nous avons observé que lorsqu'on essayait de rendre ce régulateur plus rapide, on aboutissait à des dépassements plus importants expérimentalement qu'en simulation. Les signaux sont bien sûr plus bruités sauf la vitesse de la machine qui a été filtrée avant l'acquisition, nous pensons que ces résultats sont très satisfaisants et vont nous servir de base pour les comparaisons avec les autres types de commande.

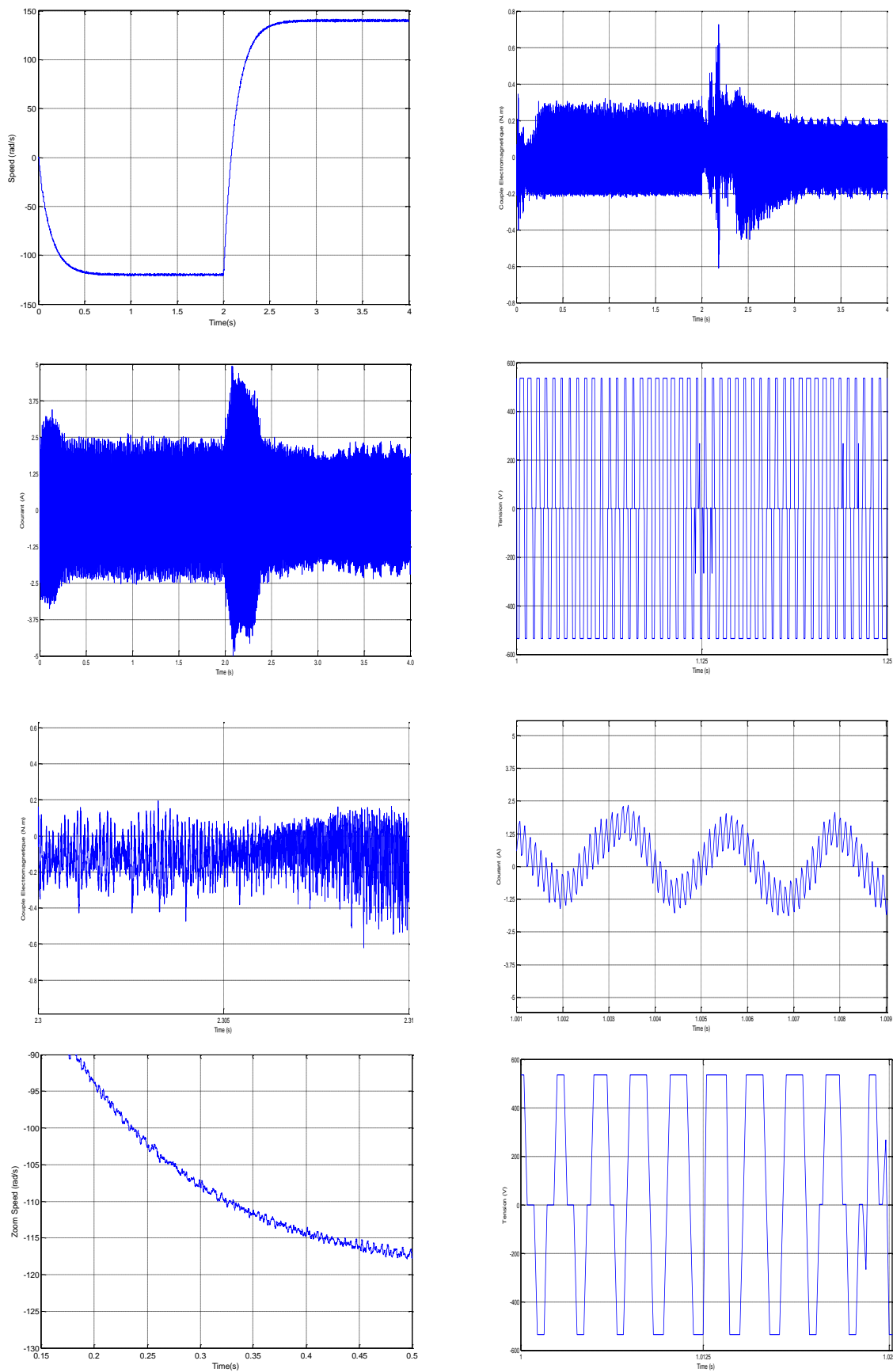


Figure 2.4 Les résultats de simulation de l'approche classique

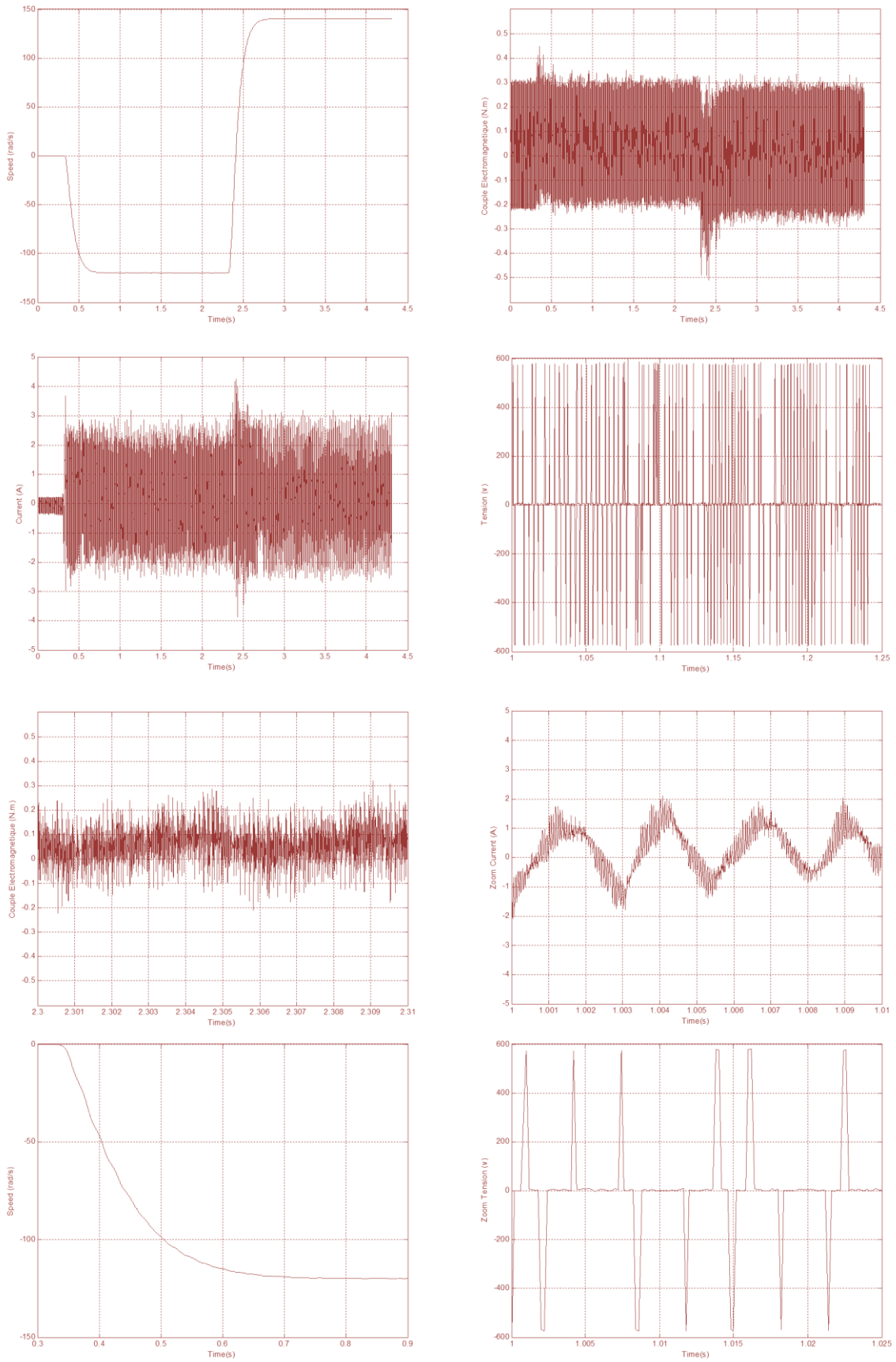


Figure 2.5 Les résultats expérimentaux de l'approche classique

2.3. Contrôle par Modèle Inverse Floue (MIF)

2.3.1 Introduction

Dans cette partie, nous allons nous intéresser au remplacement du bloc d'orientation du flux rotorique du schéma de commande vectorielle précédent par un nouveau modèle intelligent vis-à-vis le changement paramétrique de la machine, en utilisant les techniques intelligentes de la modélisation floue. Comme mentionné au chapitre 1, le nombre de combinaisons et de variantes est quasi infini. Devant ce large éventail, notre choix sera d'abord dicté par la simplicité de mise en œuvre. Nous cherchons un modèle inverse floue de la machine que l'on peut implanter au sein de la commande numérique que nous avons développée autour d'un couple de DSP, TMS et dSPACE. Une des contraintes est la limitation du temps de calcul tout en conservant les propriétés du modèle inverse flou.

2.3.2 Modèle inverse flou

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. En appliquant la modélisation floue à cette dernière nous permet donc d'obtenir le modèle inverse flou du moteur en supposant avoir la même condition d'orientation du flux.

Compte tenu de ce qui vient dit, nous avons retenu pour le modèle [43 et 47] :

- Un nombre limité à trois, cinq ou sept ensembles flous pour chaque variable.
- Des variables d'entrée dont les fonctions d'appartenance des ensembles flous sont de forme gaussiennes et sigmoïdales figure 2.8.
- Des gaussiens pour les fonctions d'appartenance de la variable de sortie figure 2.8.

L'expression du modèle de la machine asynchrone, tenant compte de la condition d'orientation de flux rotorique développer dans la première partie de ce chapitre est écrite comme suit [43] :

$$y = f(x, \eta) \quad 2.12$$

Où

$$y = (v_{ds}, v_{qs}, \omega_s)_T; x = (C_e, \frac{dC_e}{dt}, \Phi_r, \frac{d\Phi_r}{dt}, \Omega)_T \quad 2.13$$

Et

$$f = (f_1, f_2, f_3)^T \quad 2.14$$

En tenant compte de la condition d'orientation du flux rotorique et on prend Φ_r comme constant nous aurons :

$$\frac{d\Phi_r}{dt} = 0. \quad 2.15$$

Et le nouveau vecteur de variables d'état du système est [2,4,5,6 et 7]:

$$x = (C_e, \frac{dC_e}{dt}, \Phi_r, \Omega)^T \quad 2.16$$

avec

$$f_1(x) = v_{ds} = \frac{R_s}{L_m} \Phi_r - \frac{R_r(L_s L_r - L_m^2)}{L_m p^2} C_e^2 - \frac{(L_s L_r - L_m^2)}{L_m} \Omega C_e \quad 2.17$$

$$f_2(x) = v_{qs} = \frac{pL_s}{L_m} \Omega \Phi_r + \frac{(L_r R_s + L_s R_r)}{pL_m} \frac{C_e}{\Phi_r} + \frac{(L_s L_r - L_m^2)}{pL_m} \frac{1}{\Phi_r} \frac{dC_e}{dt} \quad 2.18$$

$$f_3(x) = \omega_s = \frac{R_r}{p} \frac{C_e}{\Phi_r^2} + p\Omega \quad 2.19$$

Il reste cependant à concevoir la représentation floue de l'équation (2.24) qui rassemble toutes les incertitudes inhérentes au système tel que les constantes de temps rotorique et statorique et les paramètres mécaniques; celle-ci est définie par [43 et 46] :

$$\tilde{y} = \begin{pmatrix} \tilde{v}_{ds} \\ \tilde{v}_{qs} \\ \tilde{\omega}_s \end{pmatrix} = \tilde{f}(\tilde{x}) = \begin{pmatrix} \tilde{f}_1(\tilde{C}_e, \tilde{\Phi}_r, \tilde{\Omega}) \\ \tilde{f}_2(\tilde{C}_e, \frac{d\tilde{C}_e}{dt}, \tilde{\Phi}_r, \tilde{\Omega}) \\ \tilde{f}_3(\tilde{C}_e, \tilde{\Phi}_r, \tilde{\Omega}) \end{pmatrix} \quad 2.20$$

telque; $\tilde{C}_e, \frac{d\tilde{C}_e}{dt}, \tilde{\Phi}_r, \tilde{\Omega}$ et $\tilde{v}_{ds}, \tilde{v}_{qs}, \tilde{\omega}_s$ sont les variables floues correspondant aux variables d'entrée $C_e, \frac{dC_e}{dt}, \Phi_r, \Omega$ et aux variables de commande v_{ds}, v_{qs}, ω_s respectivement.

Le modèle inverse flou est construit selon les quatre étapes de l'algorithme de la modélisation floue . Dans ce cas, le nombre d'ensembles flous des variables de sortie v_{ds}, v_{qs}, ω_s est égal à

$M_1 * M_3 * M_4$, $M_1 * M_2 * M_3 * M_4$ et $M_1 * M_3 * M_4$ respectivement. $M_i, i = 1, \dots, 4$, et le nombre de fonctions d'appartenance des variables d'entrées $C_e, \frac{dC_e}{dt}, \Phi_r, \Omega$.

Les règles sont présentées comme suit [43 et 47] :

$$\begin{aligned}
 R_{v_{ds}}^{(k_1, \dots, k_4)} : \mathbf{IF} \quad (C_e, \Phi_r, \Omega) \mathbf{is} (F_{C_e}^{k_1}, F_{\Phi_r}^{k_1}, F_{\Omega}^{k_3}) \\
 \mathbf{THEN} \quad v_{ds} \quad \mathbf{is} \quad F_{v_{ds}}^{(k_1, k_1, k_3)}
 \end{aligned} \tag{2.21}$$

$$\begin{aligned}
 R_{v_{qs}}^{(k_1, \dots, k_4)} : \mathbf{IF} (C_e, \frac{dC_e}{dt}, \Phi_r, \Omega) \mathbf{is} (F_{C_e}^{k_3}, F_{\frac{dC_e}{dt}}^{k_2}, F_{\Phi_r}^{k_1}, F_{\Omega}^{k_4}) \\
 \mathbf{THEN} \quad v_{qs} \quad \mathbf{is} \quad F_{v_{qs}}^{(k_3, k_2, k_1, k_4)}
 \end{aligned} \tag{2.22}$$

$$\begin{aligned}
 R_{\omega_s}^{(k_1, \dots, k_4)} : \mathbf{IF} \quad (C_e, \Phi_r, \Omega) \mathbf{is} (F_{C_e}^{k_1}, F_{\Phi_r}^{k_1}, F_{\Omega}^{k_3}) \\
 \mathbf{THEN} \quad \omega_s \quad \mathbf{is} \quad F_{\omega_s}^{(k_1, k_1, k_3)}
 \end{aligned} \tag{2.23}$$

La défuzzification est calculée suivant la méthode de la moyenne pondérée [43].

$$f_i = \frac{\sum_{k_1} \sum_{k_2} \sum_{k_3} \sum_{k_4} \bar{f}_i^{(k_1, k_2, k_3, k_4)} T(\mu_{x_1}^{k_1}, \mu_{x_2}^{k_2}, \mu_{x_3}^{k_3}, \mu_{x_4}^{k_4})}{\sum_{k_1} \sum_{k_2} \sum_{k_3} \sum_{k_4} T(\mu_{x_1}^{k_1}, \mu_{x_2}^{k_2}, \mu_{x_3}^{k_3}, \mu_{x_4}^{k_4})} \tag{2.24}$$

f_i représente une des variables de sortie, $i = 1, \dots, 3$.

$\bar{f}_i^{(k_1, k_2, k_3, k_4)}$ est la valeur représentative de l'ensemble flou $F_{f_i}^{(k_1, k_2, k_3, k_4)}$ qui est la sortie de la règle $R_{f_i}^{(k_1, k_2, k_3, k_4)}$..

T : opération de T-norme, qui renvoie dans notre cas le produit de ses arguments.

Le schéma bloc correspondant au modèle inverse flou ainsi conçu est représenté par la figure 2.6 [43].

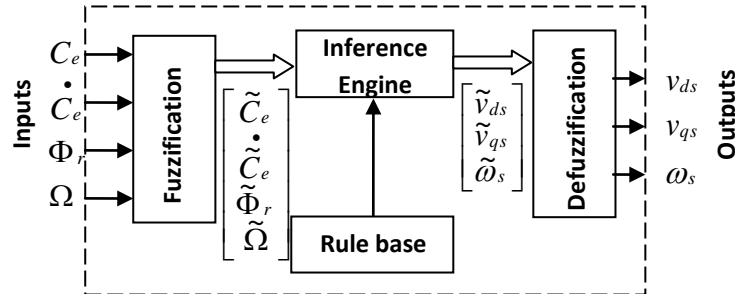


Figure. 2.6. Schéma du Modèle Inverse Flou (MIF) de la MAS

Théoriquement, nous pouvons remarquer que plus le nombre de sous-ensembles flous des entrées augmentent, 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, et l'implémentation en temps réels.

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 2.7 [43].

Afin de réduire la complexité du modèle inverse flou proposé, nous avons décomposé ce dernier en trois sous-blocs moins complexes que le précédent. Chaque bloc ayant une sortie, comme le montre la figure 2.7.

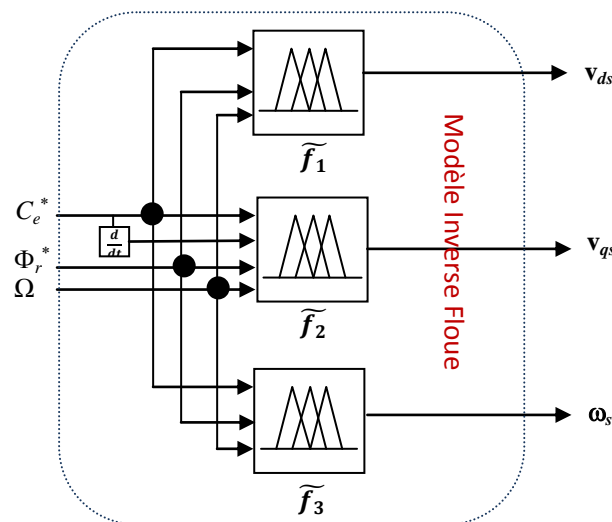


Figure 2.7 : Décomposition du modèle inverse flou

en des sous-systèmes flous

La figure 2.8 représente les fonctions d'appartenance des différentes entrées – sorties du modèle inverse flou.

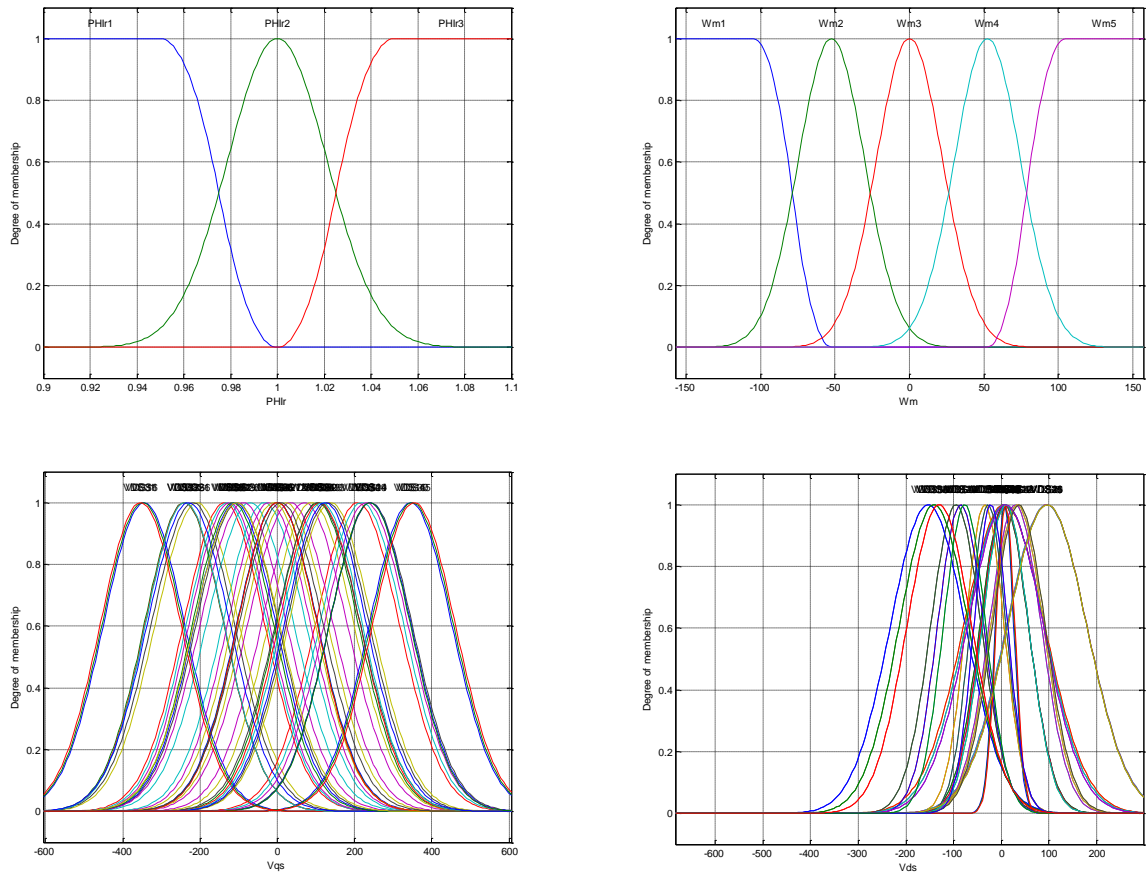
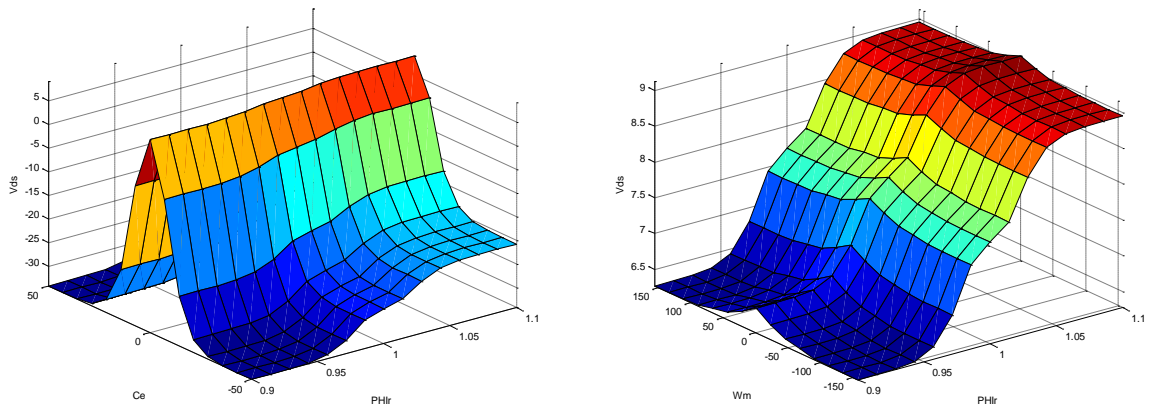


Figure 2.8 Fonctions d'appartenance Entrées – Sorties de MIF

La surface caractéristique du modèle inverse flou prend alors la forme de la figure 2.9. On observe bien sûr une symétrie apparait qui prouve le fonctionnement dans les quatre quadrants.



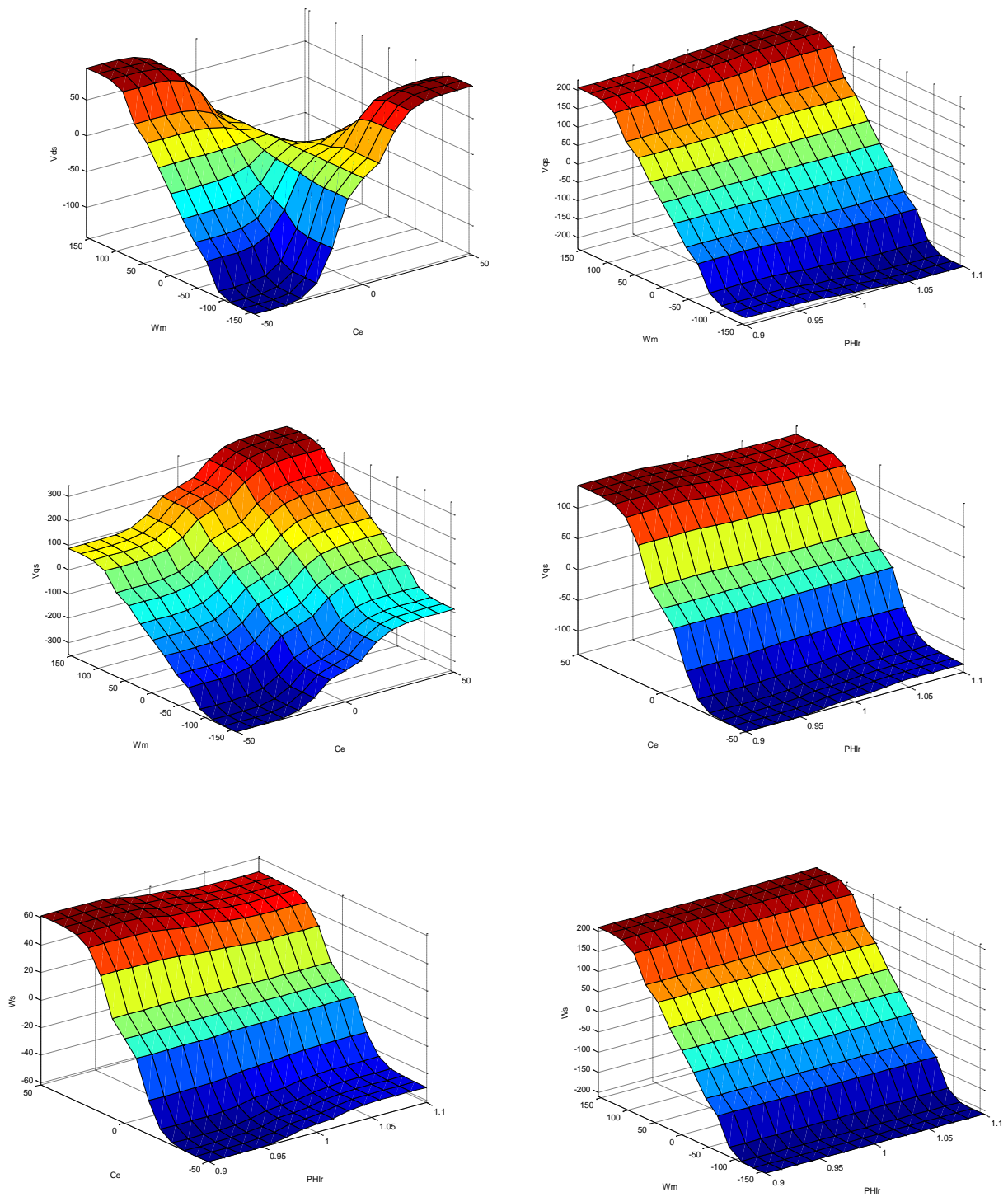


Figure 2.9 Surfaces caractéristiques du modèle inverse fluo

La figure 2.10 représente le nouveau schéma bloc de la régulation de vitesse du moteur asynchrone commandé par l'approche classique par orientation du flux rotorique en utilisant un nouveau modèle, c'est le modèle inverse flou de la machine asynchrone en tenant compte de la condition d'orientation du flux rotorique et la plage de variation des paramètres du moteur, c'est un modèle intelligent et robuste vis-à-vis des variations paramétriques. Les principaux constituants dans ce schéma sont la boucle de régulation de vitesse, et le modèle inverse flou de la machine qui donne les tensions d'alimentations et la pulsation statorique.

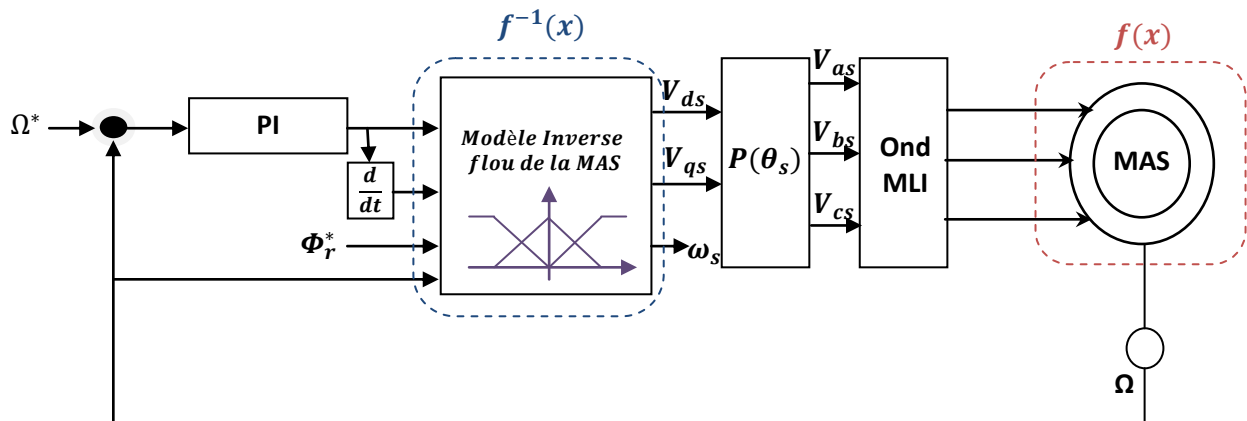


Figure 2.10 Régulation de vitesse par l'approche classique
En utilisant un modèle intelligent et robuste

Afin de valider notre modèle intelligent de la commande, des évaluations expérimentales ont été réalisées. On a utilisé les conditions d'essai que précédemment.

Des auteurs, [9,38 et 40], présentent une analyse de stabilité, appliquant la théorie de l'hyper stabilité, d'un système mono-entrée mono-sortie linéaire à paramètres constants (SISO-LTI). Ce qui n'est bien sûr pas le cas de nombreux systèmes, dont le nôtre, et pour lesquels le modèle flou est utilisé et donnent satisfaction. Le critère de Popov, sous une forme étendue, a été retenu par [9] pour leurs études de stabilité.

L'analyse rigoureuse de stabilité n'est souvent pas possible dans les cas pratiques. Elle va de plus à l'encontre du principal avantage de l'approche floue, c'est à dire son utilisation pour les systèmes qui sont difficilement modélisables [9].

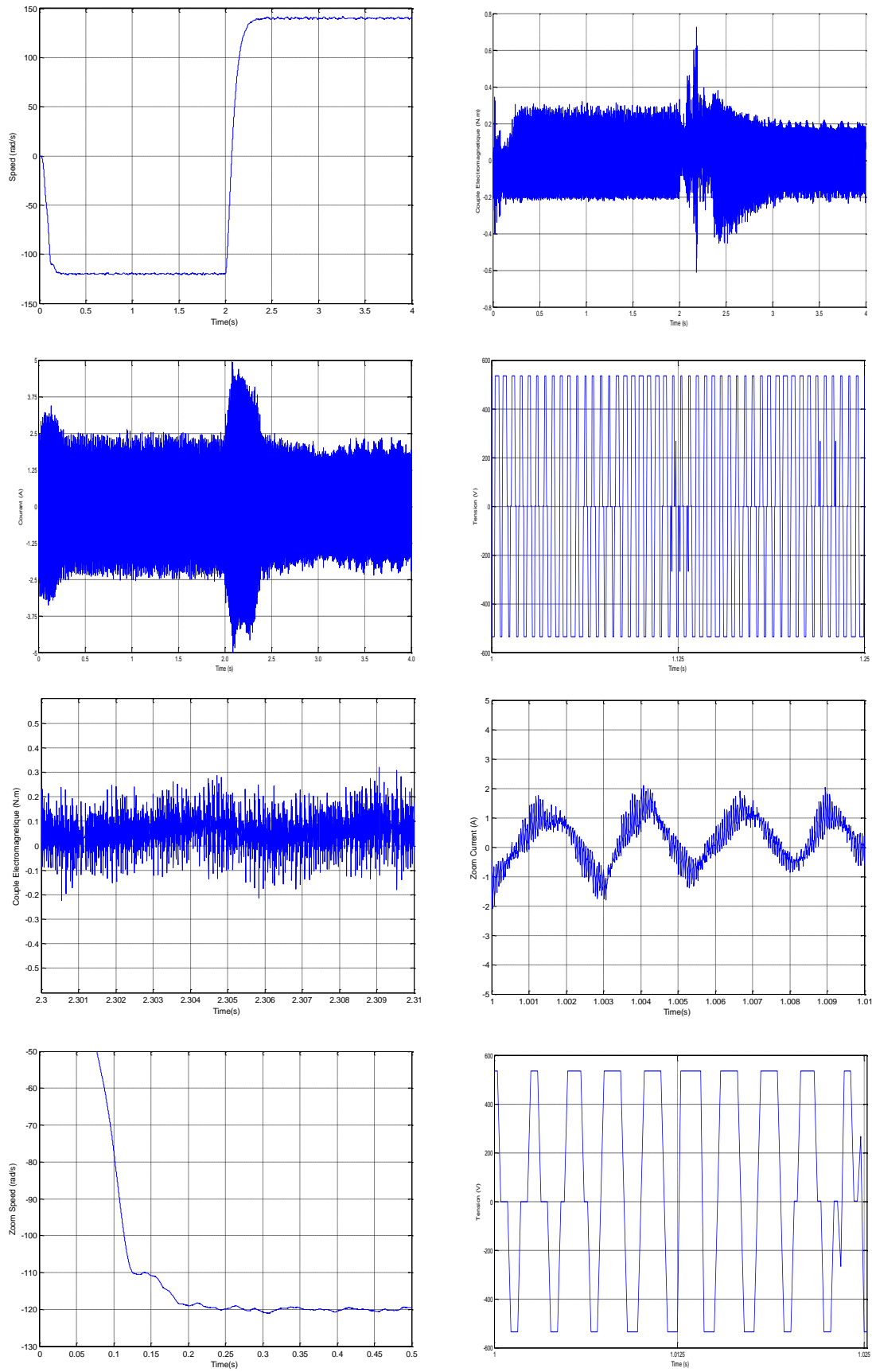


Figure 2.11 Les résultats de Simulation du MIF pour une consigne de vitesse de -120 rad/sec à 140 rad/sec

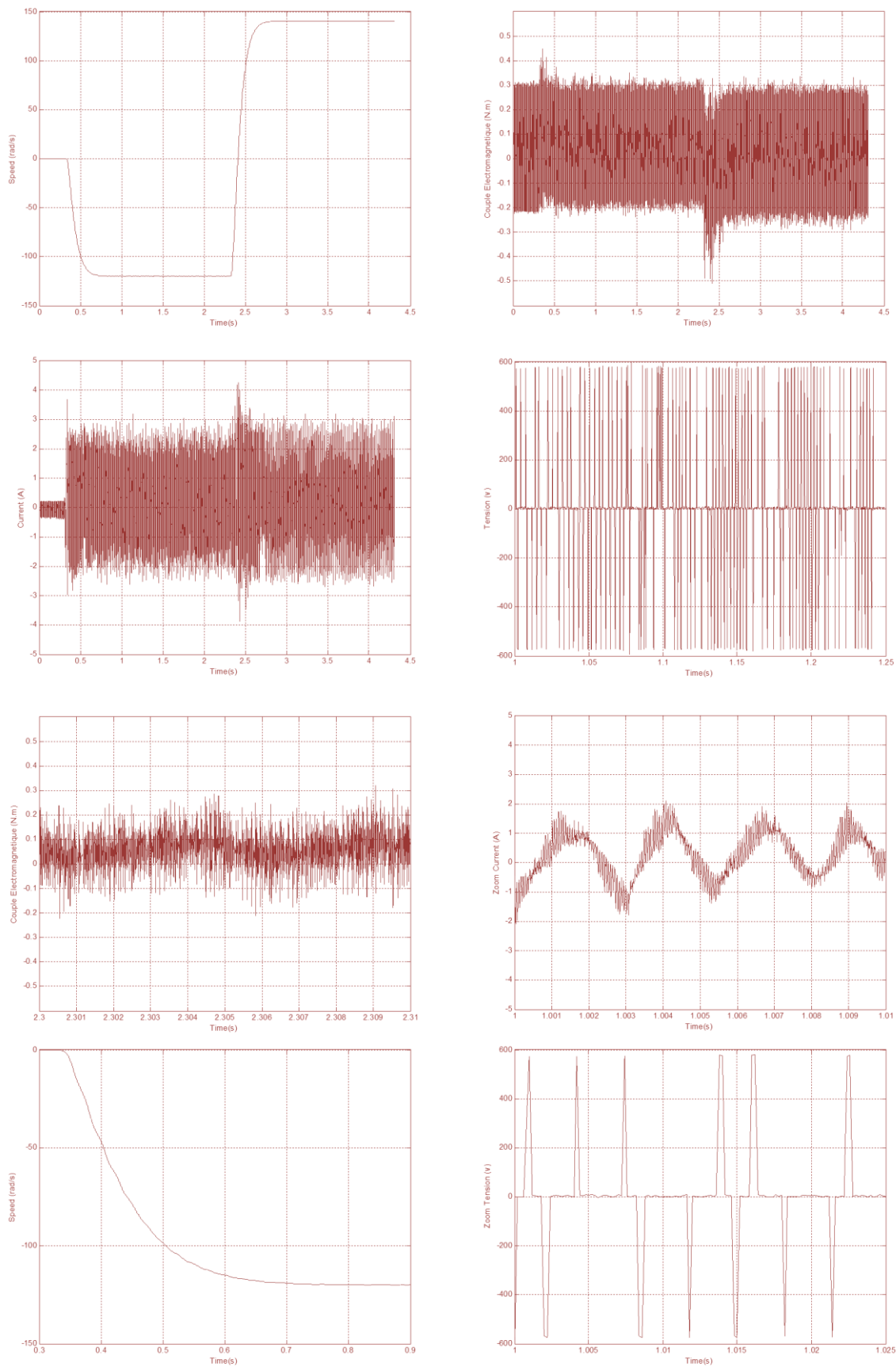


Figure 2.12 Les résultats expérimentaux du MIF pour une consigne de vitesse de -120 rad/sec à 140 rad/sec

Sur les figures 2.11 et 2.12, nous avons présenté les résultats obtenus avec le nouveau modèle intelligent pour qui conserve la condition d'orientation de flux rotorique pour contrôler la vitesse au sein du schéma de commande vectoriel classique. L'essai représente une inversion de vitesse de -120 à 140 rad/sec à vide. Les résultats expérimentaux et de simulation sont superposés et la commande marche bien avec ce nouveau modèle. On observe un temps de montée de la vitesse similaire en simulation par rapport à celui de l'essai expérimental de même que la valeur en régime permanent. Nous avons obtenu les mêmes performances de l'approche classique et ce en fonctionnement dans des conditions normales.

D'après les résultats obtenus en simulation et en expérimentation, nous remarquons que les réponses dynamiques dans ce cas ont les mêmes allures que ceux obtenues avec le CFO classique dans des conditions normales. Ce qui nous permet de dire que le modèle inverse flou adopté a pu décrire le modèle inverse classique à un degré donné de précision. Dans cette partie, nous nous sommes intéressés à la commande du moteur à induction par un modèle intelligent utilisant les avantages apportés par la modélisation tout en gardant la condition d'orientation du flux rotorique. Nous avons dans ce cas, converti la représentation analytique du bloc de découplage dans la commande à flux orienté en une représentation basée sur des règles linguistiques de type If-Then [43,44 et 46], tout en tenant compte des incertitudes et des variations paramétriques liées au moteur asynchrone.

L'approche intelligente de la modélisation floue offre 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 des ensembles flous et les formes des ensembles flous des entrées et/ou des sorties, et d'autre part d'assurer une simplicité d'implémentation en temps réels, de rapidité de calcul et de réduire la capacité mémoire. Pour notre modèle flou, nous avons utilisés des fonctions sigmoïdales et des gaussiennes comme fonctions d'appartenance pour les quatre entrées et les trois sorties du modèle, un nombre limités à trois, trois, trois et cinq fonctions d'appartenance du couple, la dérivée du couple, le flux et la vitesse respectivement, on était limité par la capacité mémoire de la carte de dSPACE DS 1103 [26] qui ne supporte pas un nombre supérieurs. L'implémentation a été très facile, et rapide, vu la simplicité apportée par le puissant logiciel de simulation MATLAB en parallèle avec le logiciel dédié avec la carte dSPACE Desktop [26 et 66].

Afin de mettre en évidence les avantages apportés par l'approche classique en utilisant le modèle inverse flou relativement à celle par le modèle mathématique, une étude de robustesse a été effectuée.

2.4 Etude de Robustesse

2.4 .1. Introduction

Les paramètres des machines ne donnent pas lieu à des valeurs certaines et définitives [9 et 43]. Ces variations correspondent à la dispersion constatée lors de la fabrication en nombre, aux évolutions des valeurs lors du fonctionnement ou aux méthodes utilisées pour l'identification des machines. Il est alors particulièrement intéressant de comparer les réponses du système commandé par l'un ou l'autre des régulateurs.

Partant des régulateurs optimisés pour une réponse avec une machine parfaitement identifiée, nous comparerons cette réponse à celles où l'on fait varier jusqu'à plus ou moins 80 % de la valeur du paramètre. Une telle gamme de variation est très large, elle permet cependant de montrer que certains paramètres n'ont qu'une influence très réduite sur la réponse du système.

Les tests portent aussi bien sur la réponse à des échelons de consignes que sur la réponse à une variation de la charge. Lors des simulations effectuées, nous amenons tout d'abord la vitesse de la machine à -120 rad/sec par l'action sur la variable "référence de vitesse", puis nous procédons à une inversion de vitesse jusqu'à +140 rad/sec.

2.4.2. Paramètres mécaniques

Il est tout à fait logique que l'influence du moment d'inertie du système tournant (machine + frein) ait une grande importance sur la réponse en vitesse [4 et 9].

2.4.2.1. Moment d'inertie (J)

Nous simulons le comportement en vitesse pour des variations de -50 % à +50 % sur l'inertie du système et procédons à une comparaison lors de l'utilisation du modèle classique figure 2.2 et le modèle inverse flou avec le même régulateur PI.

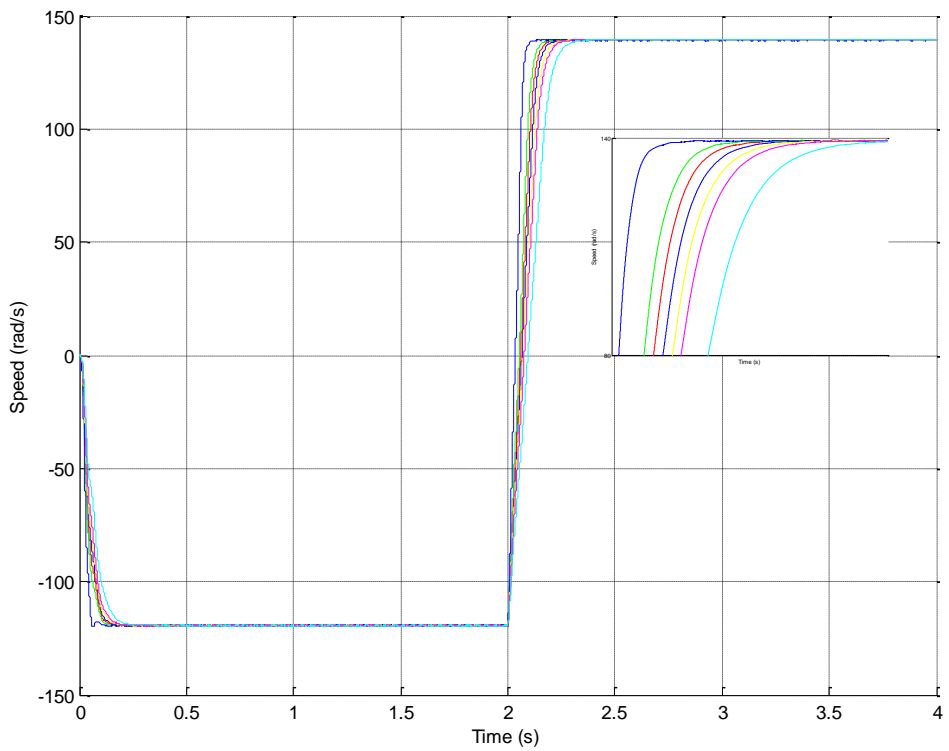
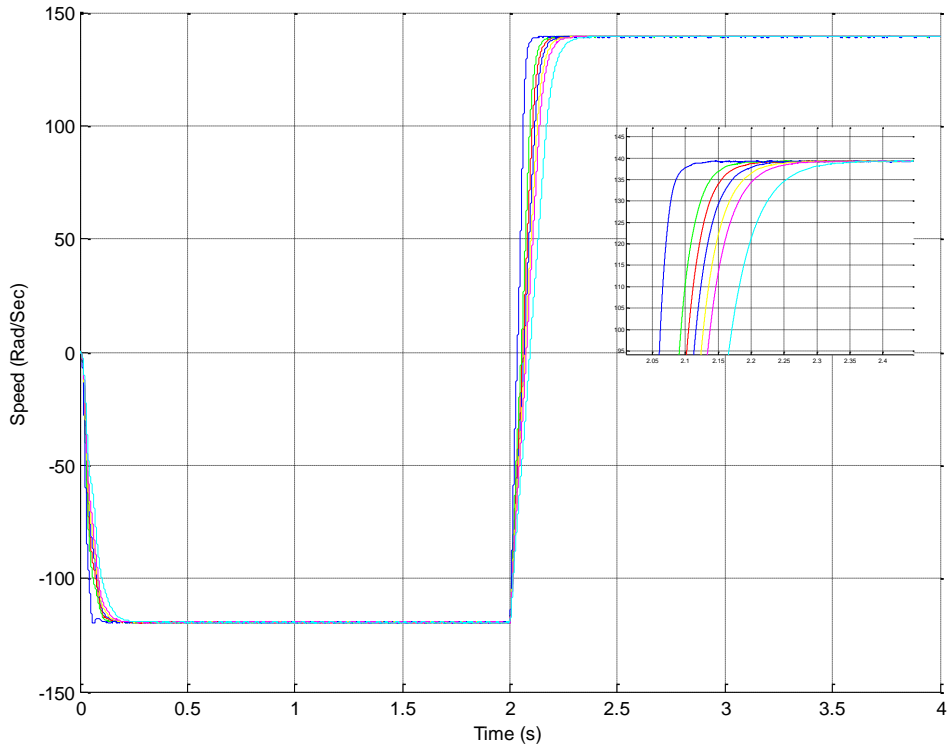


Figure 2.13 Test de Robustesse pour différentes valeurs de J de $\pm 50\%$

Afin de permettre des comparaisons moins subjectives que celles reposant sur les allures des courbes, nous évaluons les deux indices suivants [9] :

- Temps de montée : il correspond au temps que met la vitesse pour atteindre -119.6 rad/sec lors d'un échelon de -120 à $+120$ rad/sec, c'est à dire, moins de $0,33$ % de la vitesse de consigne.
- Dépassement : il correspond à l'écart entre la vitesse de consigne et la vitesse maximale atteinte lors de cet échelon.

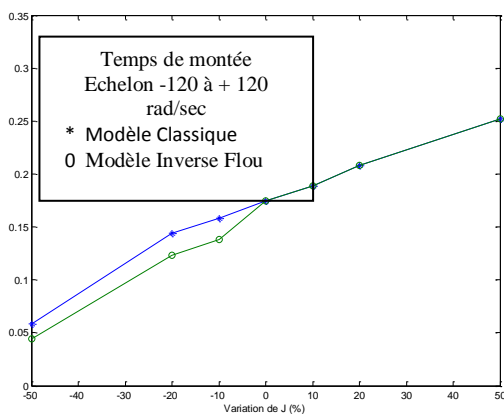


Figure 2.14 Temps de montée

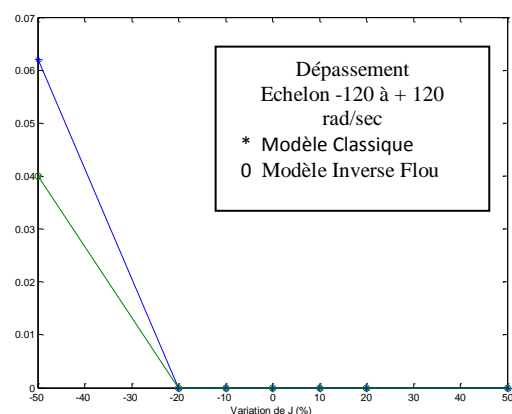


Figure 2.15 Dépassement

Il y a lieu de prêter attention tout particulièrement à l'influence de la variation du paramètre 'moment d'inertie' pour chaque modèle. En ce qui concerne la réponse en fonction du modèle avec le même régulateur, nous avons montré d'après les simulations et l'expérimentations qui ont été faites précédemment que les deux modèles ont pratiquement la même dynamique et conduisant pratiquement aux mêmes performances dans le régime normale, c.à.d. dans un fonctionnement parfait.

La nouveauté dans cette étude est que, vis à vis d'une perturbation, nous gagnerons une meilleure réponse du modèle inverse flou par rapport à celui de l'approche classique (figures 2.14 et 2.15). Par rapport au modèle mathématique classique, le nouveau modèle inverse flou offre, grâce au avantage de la modélisation intelligente par la logique flou en tenant compte des variations paramétriques dans l'élaboration du modèle, une manière différente de réglage avec notamment, la sensibilité de ce dernier par rapport aux perturbations. De ce qui précède, on

peut espérer de meilleures performances de la part de notre modèle inverse flou avec les variations des paramètres électrique.

2.4.3. Paramètres électriques

2.4.3.1 Résistance rotorique (R_r)

Les mêmes simulations sont conduites avec la résistance rotorique comme paramètre pour des variations de -50 % à +50 %.

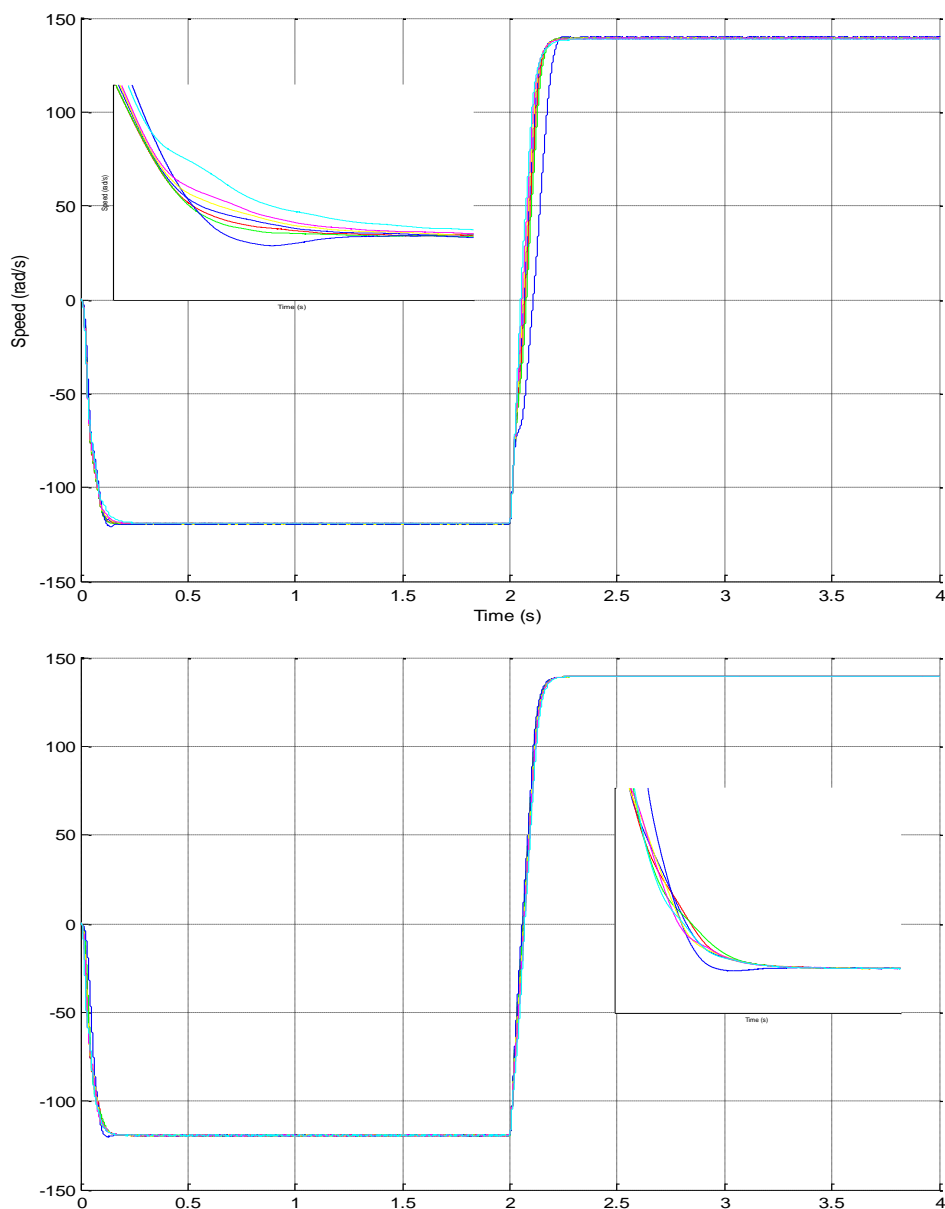


Figure 2.16 Test de Robustesse pour différentes valeurs de R_r de $\pm 50\%$

Après l'évaluation des deux indices suivants précédents à savoir, temps de montée et le dépassement nous obtenons les deux diagrammes suivants :

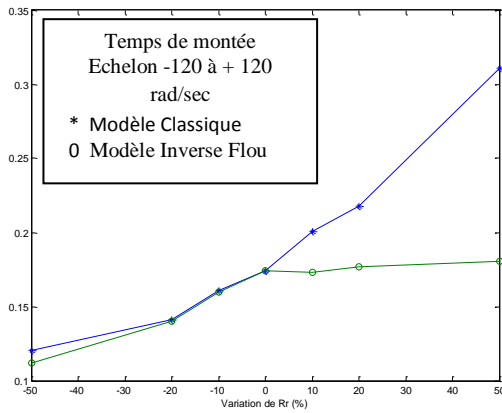


Figure 2.17 Temps de montée

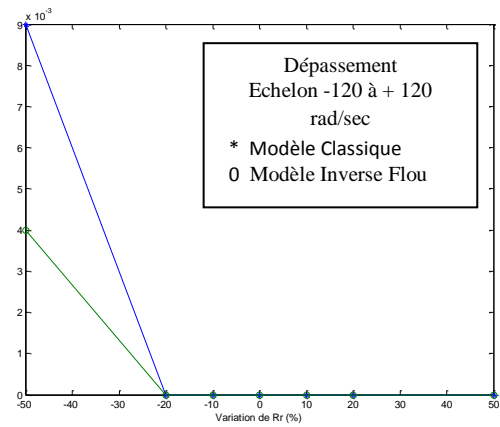
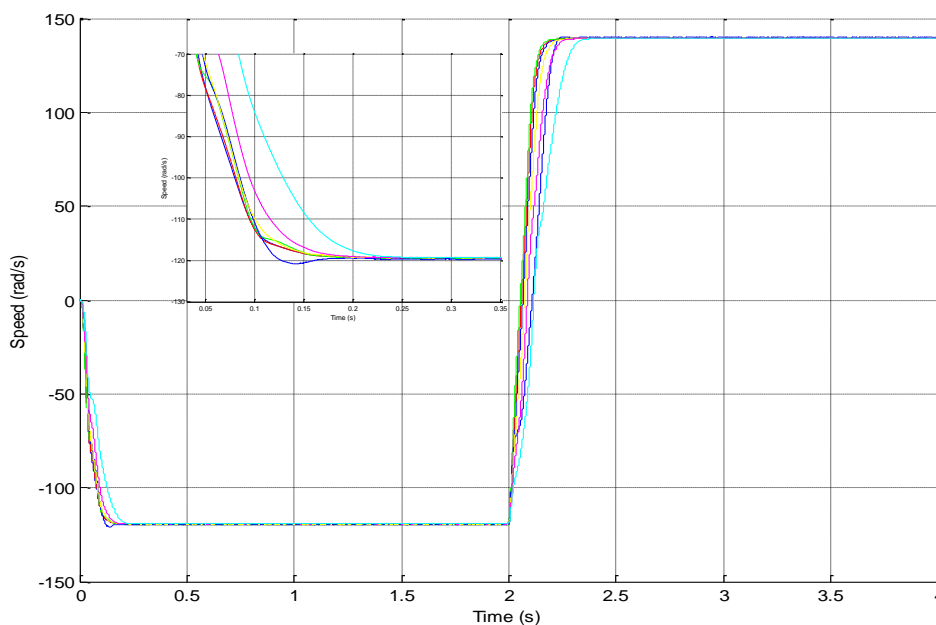


Figure 2.18 Dépassement

La robustesse de notre modèle inverse flou est encore une fois confirmée avec la variation de la résistance rotorique, on gagne en le temps de réponse en dépassement. Au vu de ces résultats, le modèle inverse flou demeure bien robuste que le modèle classique.

2.4.3.2 Inductance rotorique (L_r)

De même que les essais précédents, les mêmes simulations sont conduites avec l'inductance rotorique comme paramètre pour des variations de -20 % à +20 %.



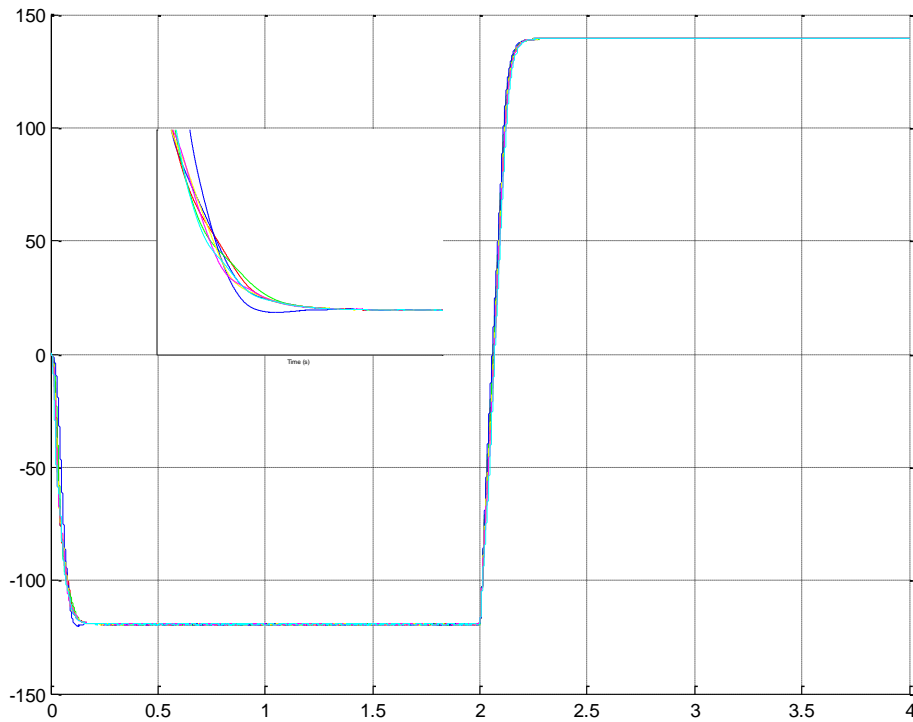


Figure 2.19 Test de Robustesse pour différentes valeurs de L_r de $\pm 20\%$

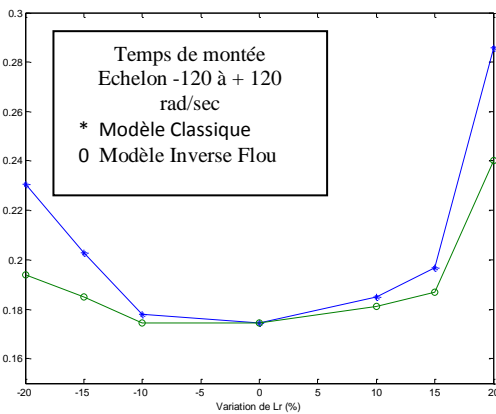


Figure 2.20 Temps de montée

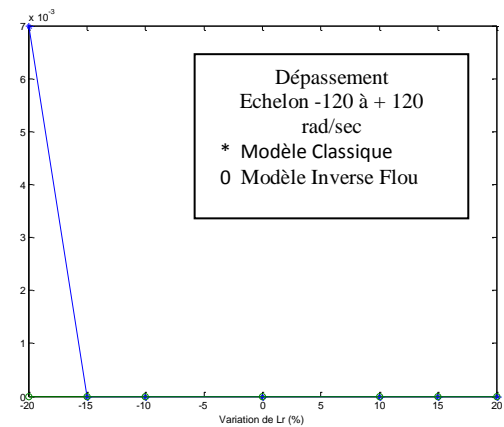


Figure 2.21 Dépassement

La sensibilité est très grande par rapport à ce paramètre. Pour les variations modérées, comprises entre -20% et $+20\%$ de la valeur de l'inductance rotorique, et cela peut s'expliquer par le fait que l'inductance rotorique ζ à d la constante du temps

rotorique utilisée dans le calcul de la pulsation statorique synchrone. Il apparaît, que le modèle inverse flou étudié conserve son comportement face à des changements de paramètres. Nous avons vérifié à travers les tests de robustesses que notre modèle inverse flou de la machine asynchrone à d'une part représentait bien son équivalent FOC classique en fonctionnement normale parfait et compense d'une autre part le principe d'orientation du flux rotorique dans des cas délicat. Ce que l'on pouvait voir à travers les résultats obtenus précédemment.

2.5 Conclusion

A travers, ce chapitre, nous avons présenté l'utilisation de la modélisation intelligente par la logique floue pour la commande de la machine asynchrone.

Nous avons présenté, une commande vectorielle de la machine asynchrone par orientation du flux rotorique. Nous nous sommes intéressés au bloc FOC d'orientation du flux rotorique, qui est le modèle inverse réduit de la machine asynchrone et à l'impact que pouvait avoir son remplacement par un modèle plus robuste surtout vis-à-vis la variation paramétriques de la machine que ce soit mécanique ou électrique sans toucher au régulateur classique PI.

Afin de pouvoir juger sur un même pied d'égalité les différents modèles, nous les avons ramenés avec le même régulateur de contrôle avec les mêmes coefficients, et pour mettre en évidence les avantages apportés par ce nouveau modèle, un test de robustesse a été fait.

Nous pouvons dire que dans un fonctionnement normale sous des conditions normales de fonctionnement c.à.d. avec connaissance parfaite des paramètres de la machine ce qui n'est pas le cas dans la pratique, le nouveau modèle inverse flou a les mêmes réponse que le modèle mathématique. Le test de robustesse nous a permet de tirer au mieux les avantages apporter par la modélisation floue en plus l'implémentation du modèle et facile.

Nous avons pu améliorer l'approche classique de commande de la machine asynchrone par orientation du flou rotorique et ce sans avoir touché au régulateur. Dans le chapitre suivant nous allons donner plus d'importance à notre modèle intelligent en ajoutant des régulateurs beaucoup plus meilleurs que le régulateur classique PID.

Chapitre 3

Chapitre 3

Application des Systèmes Evolués au Modèle Inverse Flou

3.1 Introduction

Nous avons vu précédemment l'avantage que nous a apporté la technique intelligente de la modélisation floue appliquée à la commande des moteurs électriques, en particulier le moteur asynchrone et cela sans toucher au régulateur classique PI. Afin de bénéficier plus de cet avantage qui nous a donné par ce modèle intelligent, d'autres lois de commande ont été appliquées au modèle inverse flou pour essayer de prendre 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.

3.2 Commande IP par MIF appliquée au MAS

Afin de pallier au problème de blocage par saturation de la sortie intégrale du régulateur classique proportionnel intégrale [9], un régulateur IP anti saturation est appliqué au modèle inverse flou. Le schéma de cette boucle de régulation prend alors la forme représentée par la figure 3.1.

Au lieu de bloquer tout simplement l'intégrale dès que la sortie sature, il est intéressant d'observer la structure anti-saturation (anti-windup) [9].

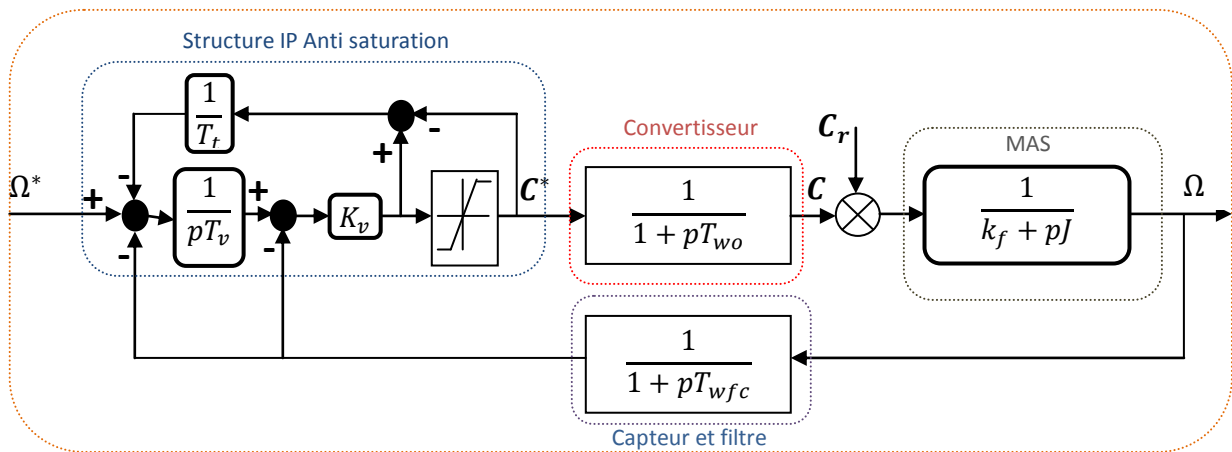


Figure 3.1 Boucle de régulation de la vitesse, structure IP anti saturation

Comme ces expressions sont très compliquées, il n'est plus possible de trouver explicitement les coefficients du régulateur adéquat. D'autant plus que les différentes limitations sur les tensions et les courants entrent en jeu dès lors que l'on procède à des échelons de consigne. Nous passons alors par des simulations afin de régler le contrôleur de vitesse. La méthode utilisée est du type essai-erreur.

Le tableau suivant résume les différents réglages obtenus:

Régulateur	Méthode d'optimisation	K_v	T_v	T_t
IP anti-saturation	Essai-erreur (Simulation)	0.78	0,015	0.7

Tableau 3.1 Réglage de contrôleur de vitesse

Les figures 3.2 et 3.3 présentent les simulations et l'expérimentation effectuées à l'aide de ce régulateur de vitesse. Après établissement du flux, nous imposons une vitesse de référence de 100 rad/sec à -120 rad/sec. Nous avons présenté l'évolution de la vitesse, le courant statorique, le couple électromagnétique et la tension composée entre phases statorique. Le courant représente bien évidemment l'image du couple électromagnétique de la sortie du régulateur de vitesse. L'anti-saturation ne rentre en jeu que lorsque le régulateur se retrouve en butée du couple C_{eref} pendant les longues phases de freinage ou d'accélération.

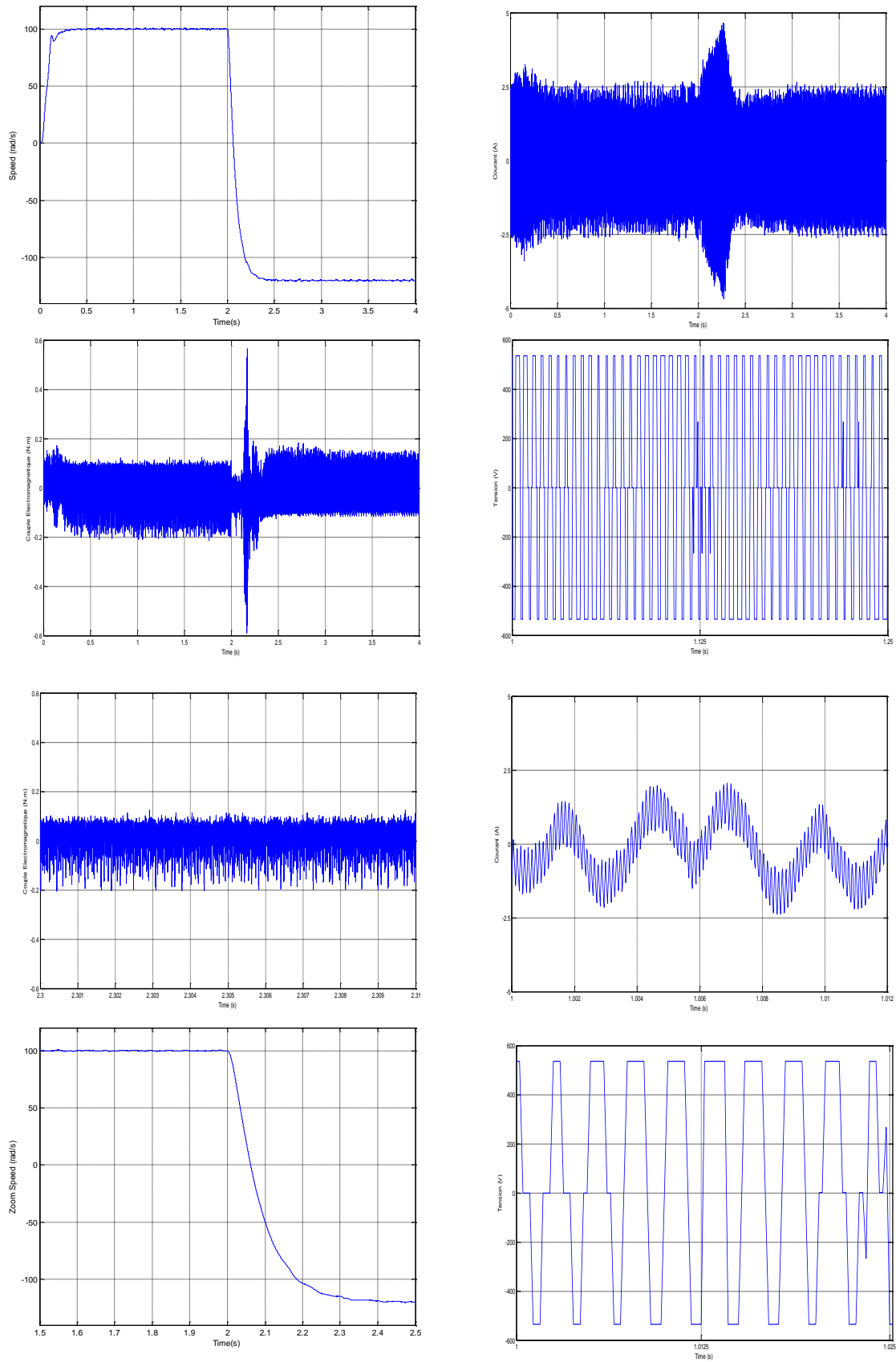


Figure 3.2 Les résultats de simulation avec un régulateur IP anti-saturation

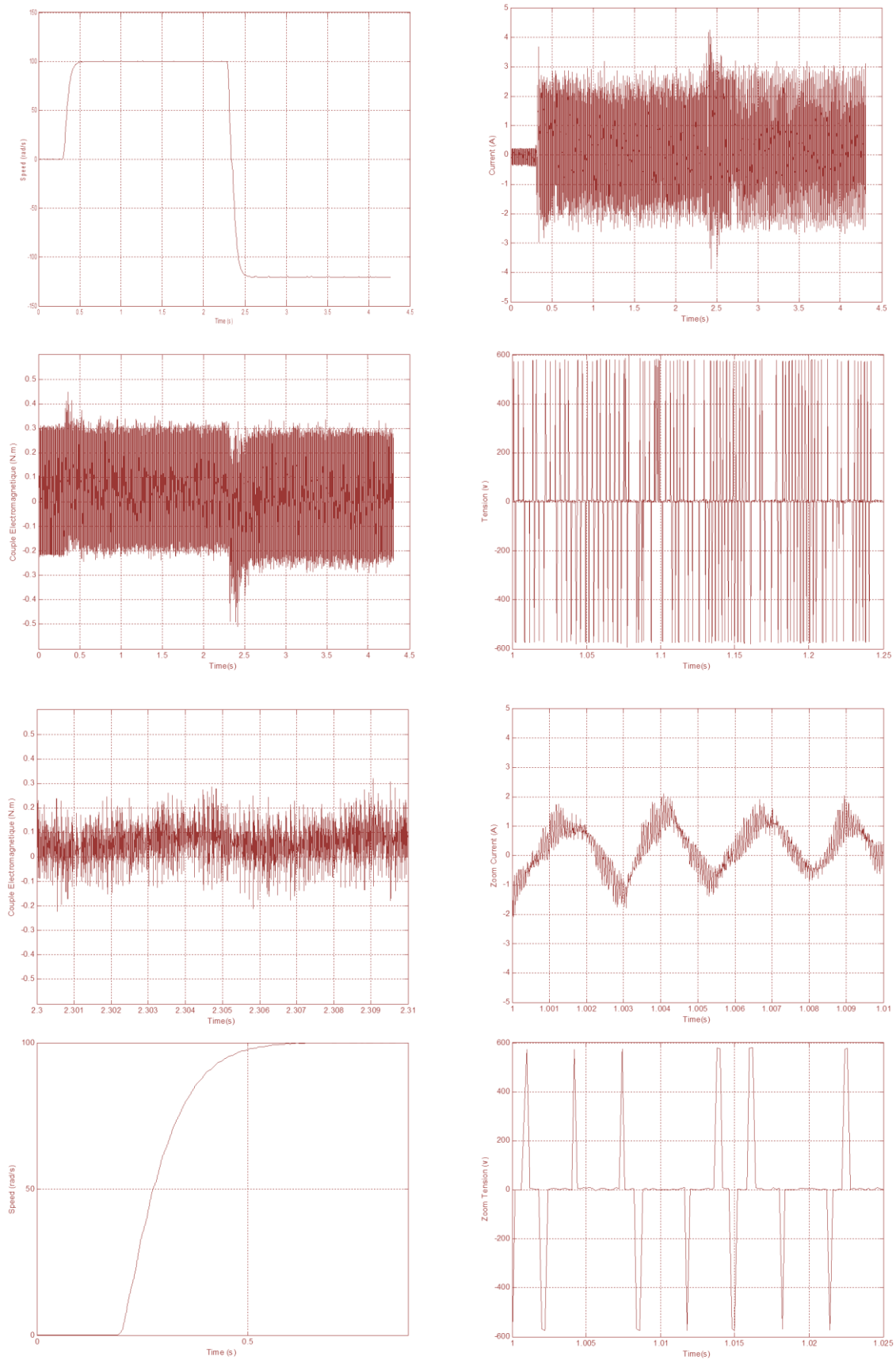


Figure 3.3 Les résultats d'expérimentation avec un régulateur IP anti-saturation

La confrontation des résultats théoriques et expérimentaux en utilisant le régulateur IP anti-saturation montre que la modélisation est suffisamment précise pour refléter les temps de réponses et les valeurs des différentes grandeurs expérimentales.

Ces figures 3.2 et 3.3 représentent une inversion de vitesse de 100 à -120 rad/sec à vide. On observe le même temps de montée. On a gagné ainsi de temps de réponse de la vitesse.

Nous avons présenté dans cette partie le régulateur IP anti saturation appliqué au nouveau modèle inverse flou retenu. Il est intéressant de remarquer que les résultats de simulation concordent avec ceux issus des essais expérimentaux. Toutefois, il n'a pas été possible d'aboutir à une méthode de calcul analytique du contrôleur de vitesse. Nous avons observé que lorsqu'on essayait de rendre ce régulateur plus rapide, on aboutissait à des dépassements plus importants expérimentalement qu'en simulation. Les signaux sont bien sûr plus bruités mais il est vraisemblable qu'il y ait un retard pur qui échappe au modèle. L'identification n'étant pas parfaite, nous pensons que ces résultats sont très satisfaisants et vont nous servir de base pour les comparaisons avec les autres types de régulateurs non linéaires.

3.3. Commande à Structure Variable Floue de la MAS basée sur le modèle inverse flou

Un contrôleur à structure variable flou est un système d'inférence flou ayant pour sortie des actions de commande flou, alors que ses entrées sont plus fréquemment considérées comme étant l'erreur et la variation de l'erreur entre une consigne et une variable du processus à contrôler. 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 [4 et 51]. 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 [4]. Dans l'approche proposée, le contrôleur flou Figure calcule l'entrée de commande floue \tilde{u} qui est le couple électromagnétique qui est le résultat du modèle flou décrit du système [4] est qui est introduite par la suite directement dans le système inverse flou sans passer la les deux étapes de fuzzification et défuzzification.

Soit le système décrit par l'équation (3.1) et par sa représentation floue suivante :

$$\tilde{x} = \tilde{f}(\tilde{x}) \mp \tilde{g}(\tilde{x}) \approx \tilde{u} \quad 3.1$$

avec :

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

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 :

$$U_e = \left\{ e \in \mathbb{R}^n / e = x_d - x ; x \in U_x \right\} \quad 3.2$$

Le vecteur d'erreur flou, $\tilde{e}(t) \in P_F(U_e)$, correspondant au vecteur $e(t)$ est donné par :

$$\tilde{e}(t) = \tilde{x}_d(t) \simeq \tilde{x}(t) = \begin{pmatrix} \tilde{e}_1(t) \\ \dots \\ \tilde{e}_n(t) \end{pmatrix} = \begin{pmatrix} \tilde{x}_{d1}(t) - \tilde{x}_1(t) \\ \dots \\ \tilde{x}_{dn}(t) - \tilde{x}_n(t) \end{pmatrix} \quad 3.3$$

avec $\tilde{e}_i(t) = \tilde{x}_{di}(t) \simeq \tilde{x}_i(t)$ 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, est décrit par :

$$U_s = U_{s1} \times U_{s2} \times \dots \times U_{sm} \subset \mathbb{R}^m \quad 3.4$$

$$U_{si} = \{ s_i(x) \in \mathbb{R} \ / \ e \in U_e \} \ ; \ i = 1, \dots, m$$

U_{si} est l'ensemble de toutes les variables possibles de la fonction de commutation s_i pour tout $e \in U_e$.

Définition 3.1 : Le vecteur flou de commutation \tilde{s} est un vecteur de m applications, défini par :

$$\tilde{s} : P_F(U_e) \rightarrow P_F(U_s) \quad 3.5$$

$$\tilde{e} \rightarrow \tilde{s}(\tilde{e})$$

avec $\tilde{s}(\tilde{e})$ l'extension de (2.6) :

$$\tilde{s}(\tilde{e}) = \begin{pmatrix} \tilde{s}_1(\tilde{e}) \\ \dots \\ \tilde{s}_m(\tilde{e}) \end{pmatrix} \quad 3.6$$

$\tilde{s}_i(\tilde{e}) \in P_F(U_{si})$ la fonction floue de commutation définie dans U_{si} et qui correspond au vecteur d'erreur \tilde{e} défini dans U_e .

Définition 3.2 : La surface floue de glissement est définie par :

$$\tilde{S}_s = \tilde{S}_{s_1} \times \cdots \times \tilde{S}_{s_m} \quad 3.7$$

$$\tilde{S}_{s_i} = \left\{ \tilde{e} \in P_F(U_e) / \tilde{s}_i(\tilde{e}) = \tilde{0} \right\}$$

avec : $\tilde{0}$ un ensemble flou constant entourant la valeur 0.

L'étendue de la dynamique désirée \dot{s}_d peut être définie par le vecteur flou :

$$\tilde{s}(\tilde{e}) = \begin{pmatrix} \tilde{s}_1(\tilde{e}) \\ \cdots \\ \tilde{s}_m(\tilde{e}) \end{pmatrix} \in P_F(U_{\dot{s}}) \quad 3.8$$

avec $P_F(U_{\dot{s}}) = P_F(U_{\dot{s}_1}) \times \cdots \times P_F(U_{\dot{s}_m})$

Définition 3.3 : La condition d'atteignabilité est définie par :

$$\exists \tilde{N} \text{ tel que } \tilde{s}(\tilde{e})^t \cdot \tilde{s}(\tilde{e}) \subset \tilde{N} \quad 3.9$$

\tilde{N} est un ensemble flou dont le support est strictement négatif.

Lemme 3.1 : La condition d'atteignabilité (3.9) est équivalente à la condition 3.10 telle que :

$$\forall s_i(e) \in \text{Supp}(\tilde{s}_i(\tilde{e})) - \{0\} \text{ et } \forall \dot{s}_i(e) \in \text{Supp}(\tilde{s}_i(\tilde{e})) - \{0\} \quad 3.10$$

$$\text{Alors : } s_i(e) \cdot \dot{s}_i(e) < 0$$

Preuve : (voir Annexe C) [4].

3.3.1 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 avant l'inversion des bases de règles, 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 :

$$P_F(U_x) = P_F(U_{x_1}) \times \dots \times P_F(U_{x_n}) \quad 3.11$$

$$P_F(U_e) = P_F(U_{e_1}) \times \dots \times P_F(U_{e_n})$$

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 :

$$F_e^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{e_1}^{k_1} \\ \dots \\ F_{e_n}^{k_n} \end{pmatrix} ; k_i = 1, \dots, M_i ; i = 1, \dots, n \quad 3.12$$

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 :

$$P_F(U_s) = P_F(U_{s_1}) \times \dots \times P_F(U_{s_m}) \quad 3.13$$

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 $P_F(U_{s_i}) = \left\{ F_{s_i}^{(k_1, \dots, k_n)} \text{ défini dans } U_{s_i} \text{ de } \mathbb{R} ; k_j = 1, \dots, M_j ; j = 1, \dots, n \right\}$

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

$$\text{les fonctions floues de commutation } \tilde{s}_i(\tilde{e}), \text{ tel que : } F_s^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{s_1}^{(k_1, \dots, k_n)} \\ \dots \\ F_{s_m}^{(k_1, \dots, k_n)} \end{pmatrix} \quad 3.14$$

avec : $F_{s_i}^{(k_1, \dots, k_n)} = \left\{ s_i(e) \in U_{s_i} \quad / \quad e \in U_e \right\}$

La base de règles correspondante à s_i est présentée par les $\prod_{i=1}^n M_i$ règles de type If-Then :

$$\begin{aligned}
 \mathfrak{R}_{s_i}^{(k_1, \dots, k_n)} : \quad & \text{IF} \quad e = (e_1, \dots, e_n)^t \quad \text{is} \quad F_e^{(k_1, \dots, k_n)} = (F_{e_1}^{k_1}, \dots, F_{e_n}^{k_n})^t \\
 & \text{THEN} \quad s_i(e) \quad \text{is} \quad F_{s_i}^{(k_1, \dots, k_n)}
 \end{aligned}
 \tag{3.15}$$

Étape 3. Choix de la dynamique floue désirée $F_{\hat{s}}^{(k_1, \dots, k_n)}$

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 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 3.10). 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 :

$$\begin{aligned}
 \text{supp}(PF_{s_i}^{(k_1, \dots, k_n)}) \cap \text{supp}(NF_{s_i}^{(k_1, \dots, k_n)}) &= \emptyset \\
 \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) &= \text{supp}(PF_{s_i}^{(k_1, \dots, k_n)}) \cup \text{supp}(NF_{s_i}^{(k_1, \dots, k_n)}) \cup \{0\} \\
 \mu(PF_{s_i}^{(k_1, \dots, k_n)}) &= \mu(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si} \quad s > 0 \quad \text{et} \quad 0 \text{ ailleurs} \\
 \mu(NF_{s_i}^{(k_1, \dots, k_n)}) &= \mu(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si} \quad s < 0 \quad \text{et} \quad 0 \text{ ailleurs}
 \end{aligned}
 \tag{3.16}$$

où $\text{supp}(F)$ représente le support de l'ensemble flou F .

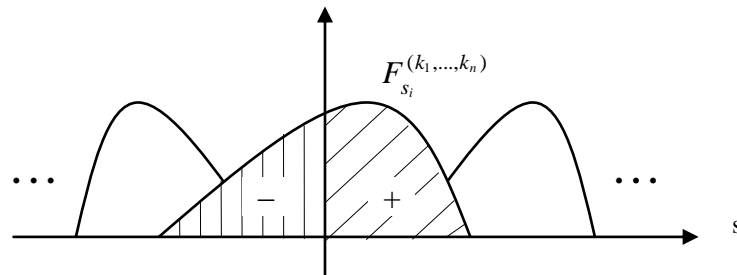


Figure 3.4 Décomposition des ensembles flous en des ensembles flous positifs et négatifs

On définit le vecteur flou \tilde{s}_d caractérisé par l'un des $\prod_{i=1}^n M_i$ vecteurs d'ensembles flous

$F_{\tilde{s}}^{(k_1, \dots, k_n)}$ tel que :

$$F_{\tilde{s}}^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{\tilde{s}_1}^{(k_1, \dots, k_n)} \\ \dots \\ F_{\tilde{s}_m}^{(k_1, \dots, k_n)} \end{pmatrix} \quad 3.17$$

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_i , et doit garantir la condition floue d'atteignabilité (3.9).

$F_{\tilde{s}_i}^{(k_1, \dots, k_n)}$ peut être défini par :

$$\begin{aligned} F_{\tilde{s}_i}^{(k_1, \dots, k_n)} &= -\tilde{M}(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si } \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \subset U_{s_i}^+ \\ &= +\tilde{M}(F_{s_i}^{(k_1, \dots, k_n)}) \quad \text{si } \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \subset U_{s_i}^- \end{aligned} \quad 3.18$$

et par :

$$F_{\tilde{s}_i}^{(k_1, \dots, k_n)} = -\omega_s \tilde{\tau} \tilde{M}(PF_{s_i}^{(k_1, \dots, k_n)}) \tilde{\tau} (1 - \omega_s) \tilde{\tau} \tilde{M}(NF_{s_i}^{(k_1, \dots, k_n)}) \quad 3.19$$

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 :

$$\begin{aligned} \omega_s &= 1 \quad \text{si } s_i \in \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \cap U_{s_i}^+ \\ &= 0 \quad \text{si } s_i \in \text{supp}(F_{s_i}^{(k_1, \dots, k_n)}) \cap U_{s_i}^- \end{aligned} \quad 3.20$$

$U_{s_i}^+ \subset \mathbb{R}^+$ 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 $\prod_{i=1}^n M_i$ règles suivantes :

$$\begin{aligned} \mathfrak{R}_{\dot{s}_i}^{(k_1, \dots, k_n)} : \quad & \text{IF} \quad e = (e_1, \dots, e_n)^t \quad \text{is} \quad F_e^{(k_1, \dots, k_n)} = (F_{e_1}^{k_1}, \dots, F_{e_n}^{k_n})^t \\ & \text{THEN} \quad \dot{s}_i(e) \quad \text{is} \quad F_{\dot{s}_i}^{(k_1, \dots, k_n)} \end{aligned} \quad 3.21$$

$$i = 1, \dots, m ; k_j = 1, \dots, M_j ; j = 1, \dots, n$$

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 (3.2) permet l'obtention de la loi de commande floue qui sera définie par :

$$\tilde{u} = i\tilde{n}v \left\{ (\tilde{s}_x(\tilde{e}))^t \sim \tilde{g}(\tilde{x}) \right\} \sim \left\{ \tilde{s}_d(\tilde{e}) \simeq (\tilde{s}_x)^t \sim \tilde{f}(\tilde{x}) \simeq \tilde{s}_t \right\} \quad 3.22$$

avec :

$$\tilde{s}_x \text{ et } \tilde{s}_t \text{ des vecteurs flous obtenus par l'extension des vecteurs } \frac{\partial s}{\partial x} \text{ et } \frac{\partial s}{\partial t}$$

respectivement.

$\tilde{f}(\tilde{x})$ 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 :

$$F_u^{(k_1, \dots, k_n)} = \begin{pmatrix} F_{u_1}^{(k_1, \dots, k_n)} \\ \dots \\ F_{u_m}^{(k_1, \dots, k_n)} \end{pmatrix} \quad 3.23$$

La base de règles correspondante à la loi de commande sera donc :

$\mathfrak{R}_{u_i}^{(k_1, \dots, k_n)}$:

$$\begin{aligned} \text{IF } x = (x_1, \dots, x_n)^t \text{ is } F_x^{(k_1, \dots, k_n)} = (F_{x_1}^{k_1}, \dots, F_{x_n}^{k_n})^t \text{ AND } s_i \text{ is } F_{s_i}^{(k_1, \dots, k_n)} \\ \text{THEN } u_i \text{ is } F_{u_i}^{(k_1, \dots, k_n)} \end{aligned} \quad 3.24$$

$i = 1, \dots, m$; $k_j = 1, \dots, M_j$; $j = 1, \dots, n$

3.3.2 Application à la commande du moteur asynchrone basée sur le Modèle Inverse Flou

3.3.2.1 Introduction

Le modèle adopté du moteur asynchrone est représenté sous une forme canonique et correspond aux équations (2.32)-(2.33) du deuxième chapitre. Comme il a été mentionné, le système (2.33) dépend étroitement des paramètres de la machine et du couple résistant, ce qui nous a incité à le commander par l'intermédiaire d'un modèle inverse flou. Pour tirer d'avantage de cette technique, une conception d'un régulateur flou robuste a été faite figure 3.5.

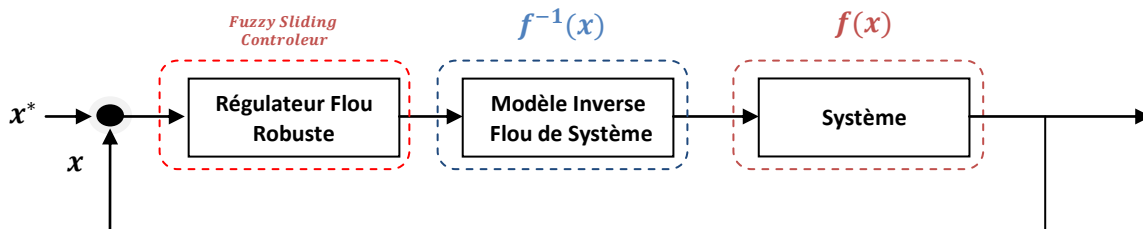


Figure 3.5 Principe de base de la commande par mode de glissement flou

3.3.2.2 Elaboration de la commande floue robuste

D'une manière générale, plusieurs contrôleurs flous peuvent être élaborés. Etant donné que le schéma de commande est à structures variables [4], nous avons opté pour la conception d'un contrôleur flou robuste ; pour cela, les étapes à suivre sont les suivantes [51 et 4]:

Étape 1. Construction la variable d'erreur floue

La variable d'erreur $e(t)$ est définie comme:

$$e = \Omega_{ref} - \Omega \quad 3.25$$

Ω_{ref} est choisie constante.

La variable d'erreur floue \tilde{e} :

$$\tilde{e} = \widetilde{\Omega_{ref}} - \tilde{\Omega} \quad 3.26$$

avec $\tilde{\Omega}_{ref}$ est un singleton flou.

La partition floue $P_F(U_{e_i})$ de l'univers de discours U_{e_i} de la variable d'entrée est construite comme suit :

$$P_F(U_{e_i}) = \left\{ F_{e_i}^{k_i} = \tilde{z}_{ref i} - F_{z_i}^{k_i} ; k_i = 1, \dots, M_i \right\} \quad 3.27$$

où $F_{e_i}^{k_i}$ ($k_i = 1, \dots, M_i$) est un ensemble flou que peut prendre, dans le présent algorithme, les variables floues \tilde{e}_1 .

Étape 2. Construction des fonctions floues de commutation

L'extension floue de la fonction de commutation $s_1(e_1)$ peut être écrite comme suit :

$$\tilde{s}_1(\tilde{e}_1) = \lambda_1 \tilde{e}_1 \quad 3.28$$

L'ensemble flou que peut prendre la fonction floue de commutation est :

$$F_{s_1}^{k_1} = \lambda_1 \tilde{e}_1 \quad 3.29$$

Étape 3. Choix de la dynamique floue désirée

Durant le mode d'atteignabilité, la robustesse est assurée par le choix de la dynamique désirée \tilde{s}_{d1} . Cette dernière est une extension de la dynamique désirée \dot{s}_{di} définie par l'équation (3.24).

L'ensemble flou $F_{\dot{s}_i}^{k_i}$ correspondant à \tilde{s}_{di} , est choisi en se référant à l'extension de la fonction non-linéaire $M_1(s_1)$.

$$F_{\dot{s}_i}^{(k_i)} = -\omega_{s_i} \tilde{\cdot} F_{M(PF_{s_i})}^{(k_i)} \tilde{+} (\omega_{s_i} - 1) \tilde{\cdot} F_{M(NF_{s_i})}^{(k_i)} \quad 3.30$$

avec

$$\omega_{s_i} = \begin{cases} 1 & \text{si } s_i \in \text{supp}(F_{s_i}^{k_i}) \cap U_{s_i}^+ \\ 0 & \text{si } s_i \in \text{supp}(F_{s_i}^{k_i}) \cap U_{s_i}^- \end{cases} \quad 3.31$$

$U_{s_i}^+ \subset \mathbb{R}^+$ 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 $\tilde{s}_i = PF_{s_i}^{(k_i)}$.

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 :

$$\tilde{v}_1 = \left(-\tilde{f}_1(\tilde{z}_1) \tilde{+} \tilde{z}_{ref1} \tilde{+} \frac{1}{\lambda_1} \tilde{\cdot} \tilde{s}_{d1}(\tilde{e}_1) \tilde{+} \frac{1}{\lambda_1} \tilde{\cdot} \tilde{e}_1 \right) \tilde{\cdot} \tilde{g}_1(\tilde{z}_1) \quad 3.32$$

Cette équation nous permet de déduire l'ensemble flous $F_{u_i}^{(k_1, k_2)}$, tels que :

$$F_{v_1}^{(k_1, k_2)} = \left(-F_{f_1}^{(k_1)} \tilde{+} \tilde{z}_{ref1} \tilde{+} \frac{1}{\lambda_1} \tilde{\cdot} F_{\dot{s}_{d1}}^{(k_1)} \tilde{+} \frac{1}{\lambda_1} \tilde{\cdot} F_{e_1}^{(k_1)} \right) \tilde{\cdot} F_{g_1}^{(k_1)} \quad 3.33$$

et la base de règles correspondant à l'entrée de commande u pour chaque sous-système :

$$\mathfrak{R}_{v_1}^{(k_1)} : \quad \text{IF} \quad (z_1, e_1)^t \quad \text{is} \quad (F_{z_1}^{k_1}, F_{e_1}^{k_1})^t$$

$$\quad \quad \quad \text{THEN} \quad v_1 \quad \text{is} \quad F_{v_1}^{(k_1)} \quad \quad \quad 3.34$$

La structure globale de la commande du moteur à induction, à structure variable floue basé sur le modèle inverse flou est la suivante (figure 3.6).

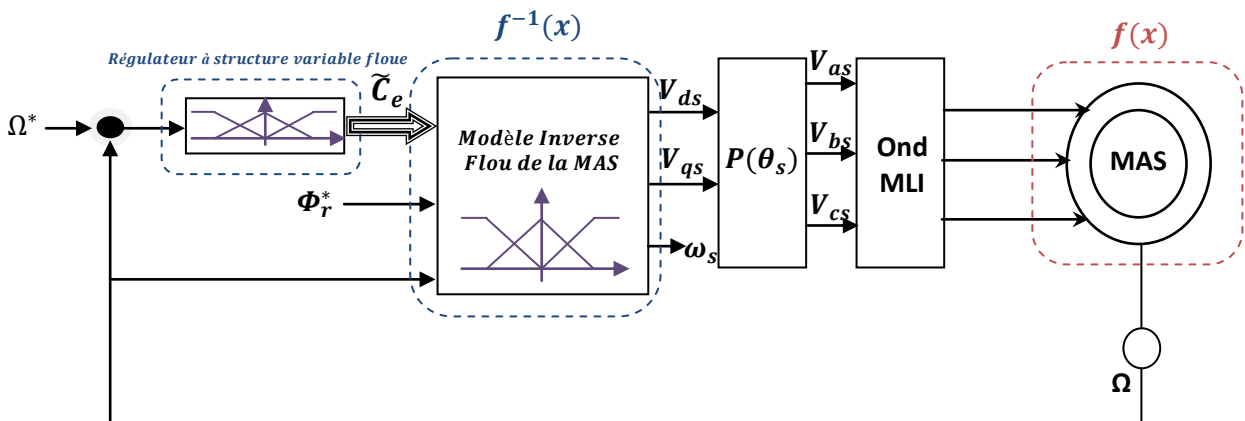


Figure 3.6 Commande à structure variable floue basé sur un modèle inverse flou

Le contrôleur flou basé sur l'approche par mode glissant fournit l'information robuste floue au modèle flou pour que ce dernier fournisse la commande à la machine asynchrone.

Afin de présenter l'avantage de cette commande une simulation numérique a été faite, vu la limitation de la mémoire dédiée sur la carte dSPACE DS 1103, on n'a pas pu faire l'implémentation réel. Un démarrage à vide à une valeur de consigne de 120 rad/s puis une inversion de vitesse de -120 rad/s a été appliquée, une charge de 10 N.m a été appliqué entre les instants 1 et 1.5 secondes. Les performances ont été nettement meilleures.

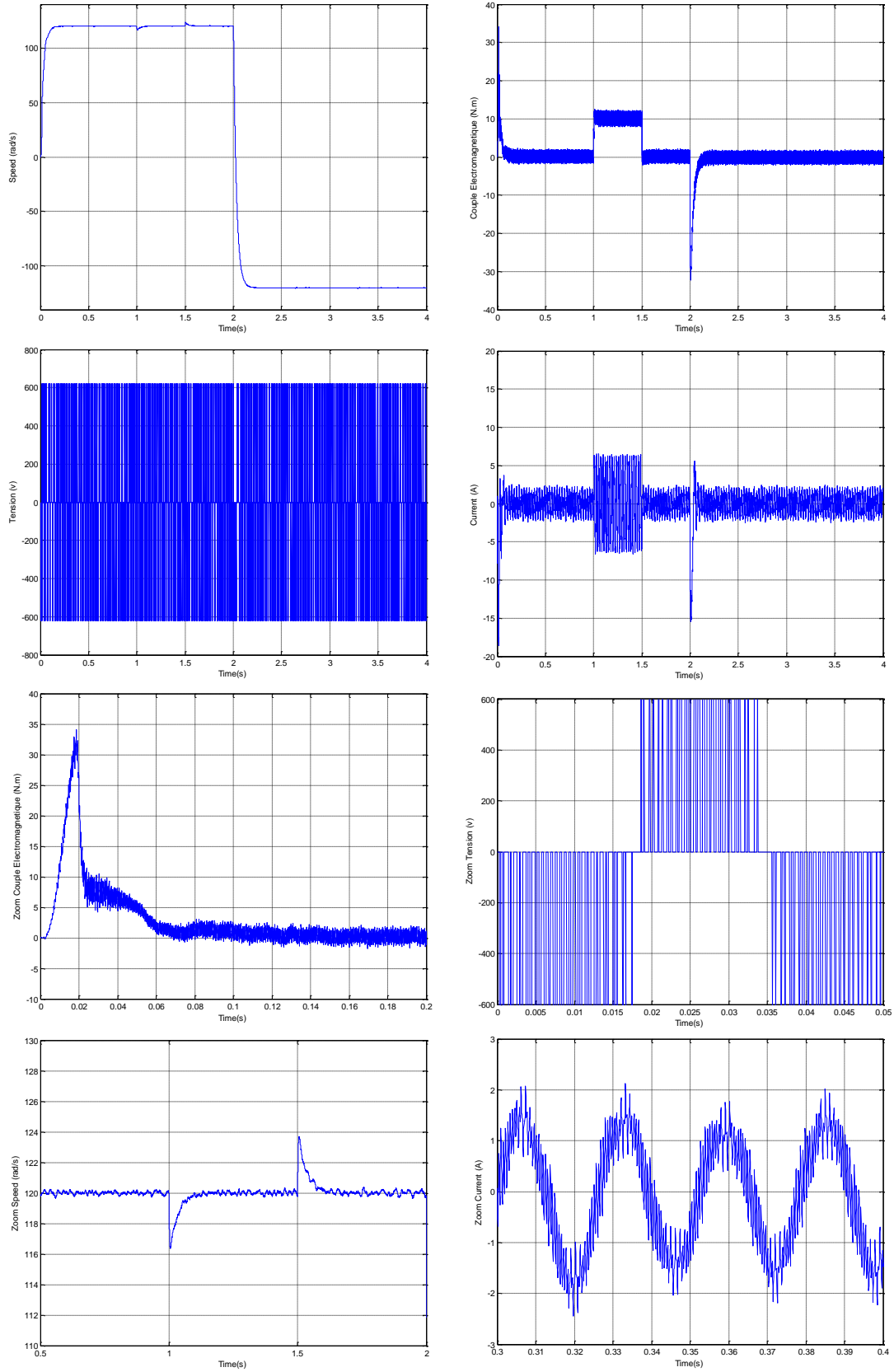


Figure 3.7 Les résultats de simulation de la commande à structure variable floue basé sur le modèle inverse flou

Les résultats de simulation montrent la dynamique de réglage de la vitesse et le rejet de toute perturbation. Notons que, dans tous les cas, la variation du couple de charge n'affecte pratiquement pas les autres variables en régime permanent, ce qui nous permet d'affirmer le maintien de la condition d'orientation du flux par le modèle inverse flou.

L'application de la commande par mode glissant flou, au modèle inverse flou pour la commande de la machine asynchrone a permis de mettre en évidence la fiabilité de cette technique. Dans cette partie, nous avons présenté une façon d'introduire la logique floue dans la commande par mode glissant de la machine asynchrone. Elle permet à cette technique d'exploiter la connaissance préalable du régulateur par mode glissant classique décrivant grossièrement le comportement du celui-là. Cette technique a donné de très bons résultats, de point de vue de la poursuite des références imposés.

3.4 Commande Adaptative Floue de la MAS basée sur le Modèle inverse flou

3.4.1 Introduction

La commande adaptative est une technique qui concerne à identifier un processus et son environnement à partir de quelques mesures entrées-sorties, et d'utiliser ensuite le modèle obtenu pour concevoir une action de commande. Depuis le régulateur auto ajustable de Procyk et Mamdani [4], plusieurs systèmes adaptatifs flous ont été élaborés et plusieurs résultats ont été publiés [25,57 et 58]. Les systèmes adaptatifs flous fournissent l'avantage d'être capables d'incorporer les informations numériques et les informations linguistiques en même temps pendant les étapes de construction et d'apprentissage. Il a été démontré que les systèmes flous peuvent approximer n'importe quelle fonction non linéaire continue à n'importe quelle précision et cela par le théorème de l'approximation universelle. Récemment, plusieurs travaux de commande adaptative floue ont été proposés [25 et 58], et leurs stabilités ont été élaborées par la méthode de Lyapunov.

A ce jour, tous les contrôleurs flous (FLC) sont basés sur le traditionnel contrôleur flou type-1. Le problème majeur d'un contrôleur flou type-1 est son incapacité de prendre en charge complètement les incertitudes numériques et linguistiques dans les systèmes inconnus, incertains, perturbés et non linéaires puisqu'il utilise des ensembles flous de type-1 précis. L'incertitude d'un certain système inconnu cause des problèmes dans la détermination exacte des fonctions d'appartenance pour les conséquences et pour les prémisses pendant l'étape de conception du FLC, et cela peut bien sur causer une dégradation dans les performances du FLC. Un système flou type-2 est un système dans lesquels les fonctions d'appartenance prémisses ou conséquences sont des ensembles flous type-2. Le concept des ensembles flous type-2 a été introduit par Zadeh [10,11,12,13 et 54], et a été récemment intensivement étudié par Mendel [11,56 et 60]. Un ensemble flou type-2 est une extension du concept de l'ensemble flou type-1 (voir annexe B). Un ensemble flou type-2 possède des degrés d'appartenance qui sont eux-mêmes flous [59,60,61 et 25]. Un degré d'appartenance type-2 peut être n'importe quel sous ensemble défini sur l'appartenance primaire $[0, 1]$; et, à chaque appartenance primaire correspondra une appartenance secondaire (qui peut être aussi dans $[0, 1]$) définissant les

possibilités des appartenances primaires [25]. Assez souvent, la connaissance utilisée pour la construction des règles dans les systèmes FLC est incertaine. Cette incertitude nous mène à obtenir des règles possédants des prémisses ou des conséquences incertaines, ce qui va être traduit par des fonctions d'appartenance prémisses et conséquences incertaines. Par conséquent, un système flou type-2 sera très efficace dans les situations où il est difficile de déterminer exactement les fonctions d'appartenance, alors, on peut dire qu'un système flou type-2 nous permet d'incorporer facilement les incertitudes des règles, ce qui va affecter bien sûr la sortie du système flou [25].

Dans ce chapitre nous proposons une méthode de conception d'une structure de commande adaptative basée sur la logique floue type-2 pour une certaine classe de systèmes dynamiques non linéaires inconnus [25,59 et 60]. Les paramètres libres du contrôleur flou type-2 seront ajustés en temps réel par une loi d'adaptation basée sur la synthèse de Lyapunov. On conçoit aussi un contrôleur de supervision en cascade avec le régulateur flou type-2. Si le système non linéaire tend à être instable par le régulateur flou type-2, spécialement dans les régimes transitoires ou lors de la présence des perturbations, la commande de supervision sera activée pour aider le contrôleur flou type-2 à stabiliser le système. Par contre, si le contrôleur flou type-2 fonctionne avec succès, alors le contrôleur de supervision sera désactivé. En terme d'effort de commande (énergie), ceci est une méthodologie économique très efficace. Par conséquent, notre structure adaptative indirecte garantira la stabilité globale du système en boucle fermée (tous les signaux dans le système sont uniformément bornés).

Formulation du problème

Soit un système dynamique non linéaire de la forme

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \\ \vdots \\ \dot{x}_n = f(x_1, x_2, \dots, x_n) + g(x_1, x_2, \dots, x_n).u + d \\ y = x_1 \end{cases} \quad 3.35$$

Ou f et g sont des fonctions réelles continues et inconnues, $u, y \in \mathbb{R}$ sont l'entrée et la sortie du système, respectivement, et d une perturbation externe bornée. Nous supposons que le vecteur d'état $\underline{x} = (x_1, x_2, \dots, x_n)^T = (x, \dot{x}, \dots, x^{(n-1)})^T \in \mathbb{R}^n$ est disponible pour la mesure. La controlabilité de ce dernier exige que $g(\underline{x}) \neq 0$ pour tous \underline{x} dans une certaine région de

controlabilité $U_c \subset \mathbb{R}^n$. Puisque $g(\underline{x})$ est continue, nous allons supposer que $0 < g(\underline{x}) < \infty$ pour tous $\underline{x} \in U_c$. Nous allons aussi supposer que les fonctions f et g sont bornées. L'objectif de commande est de trouver une loi de commande $u = \hat{u}(\underline{x})$ forçant l'état $\underline{x}(t)$ à suivre une trajectoire de référence désirée bornée $\underline{y}_m(t) = (y_m(t), \dot{y}_m(t), \dots, y_m^{(n-1)}(t))^T$.

Dans la conception des systèmes de commande adaptative, la structure du contrôleur est basée sur le modèle du procédé. Si ce dernier est inconnu, il sera alors raisonnable de le remplacer par un modèle estimé, et l'utiliser ensuite pour la conception du régulateur. Si la dynamique du procédé est connue, c'est-à-dire, f et g sont connus et que le système est libre de la perturbation externe d , nous pouvons résoudre le problème de commande par ce qu'on appelle feedback-linéarisation [25]. Dans cette méthode, f et g sont utilisées pour construire la loi de commande suivante :

$$u = u(\underline{x}) = \frac{1}{g(\underline{x})} [-f(\underline{x}) + \dot{y}_m^{(n)}(t) + \underline{k}^T e] \tag{3.36}$$

Où $e = e(t) \triangleq y_m(t) - y(t)$ est l'erreur de poursuite, $\underline{e} = \underline{e}(t) \triangleq (e, \dot{e}, \dots, e^{(n-1)})^T$, et $\underline{k} \triangleq (k_n, \dots, k_2, k_1)^T \subset \mathbb{R}^n$ est choisi tel que tous les racines du polynome

$h(s) \triangleq s^n + k_1 s^{n-1} + \dots + k_n$ se trouvent dans le demi plan gauche de l'espace complexe. En appliquant la loi de commande (3.36) au système (3.35) nous obtenons la dynamique de l'erreur suivante :

$$e^{(n)} + k_1 e^{(n-1)} + \dots + k_n e = 0$$

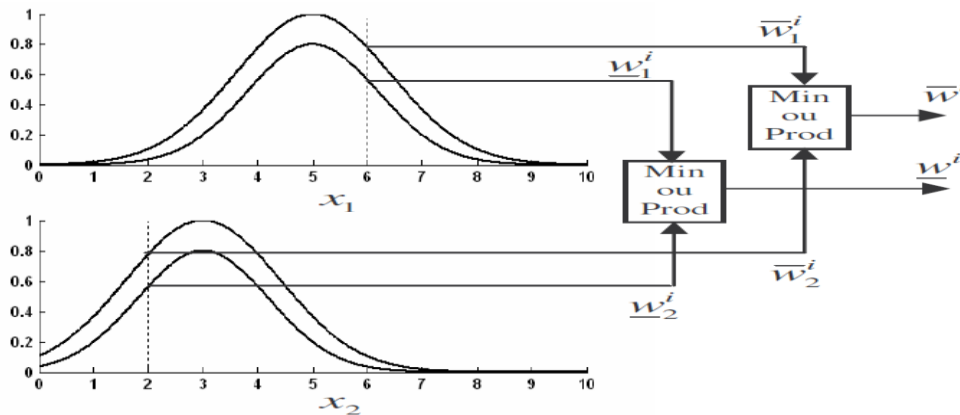


Figure. 3.8 Opérations d'entrée sur des ensembles flous type-2

Où l'objectif de commande est d'avoir $\lim_{n \rightarrow \infty} e(t) = 0$. Puisque les fonctions f et g sont supposées inconnues, nous ne pouvons pas alors les utiliser pour la construction de la loi de commande (3.36). Par conséquent, dans la suite, nous allons les remplacer par leurs estimées \hat{f} et \hat{g} , donc la loi de commande devient :

$$u_c = u_c(\underline{x}/\underline{\theta}_f, \underline{\theta}_g) = \frac{1}{\hat{g}(\underline{x}/\underline{\theta}_g)} [-\hat{f}(\underline{x}/\underline{\theta}_f) + y_m^{(n)}(t) + \underline{k}^T e] \quad 3.37$$

Où $\underline{\theta}_f, \underline{\theta}_g$ sont les paramètres libres des systèmes estimateurs \hat{f} et \hat{g} respectivement.

3.4.2. Structure de commande proposée

La structure proposée est constituée par un contrôleur adaptatif flou type-2 [25] en cascade avec un autre contrôleur de supervision. En appliquant (3.37) à (3.35) et après quelques manipulations, nous obtenons alors l'équation d'erreur suivante :

$$e^{(n)} = -\underline{k}^T \underline{e} + [\hat{f}(\underline{x}/\underline{\theta}_f) - f(\underline{x})] + [\hat{g}(\underline{x}/\underline{\theta}_g) - g(\underline{x})]u_c - d \quad 3.38$$

Qui peut être écrite dans l'espace d'état par

$$\dot{\underline{e}} = A_c \underline{e} + \underline{b} [\hat{f}(\underline{x}/\underline{\theta}_f) - f(\underline{x})] + [\hat{g}(\underline{x}/\underline{\theta}_g) - g(\underline{x})]u_c - d \quad 3.39$$

Où

$$A_c = \begin{bmatrix} 0 & 1 & \dots & 0 & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 0 & 1 \\ -k_n & -k_{n-1} & \dots & \dots & \dots & -k_1 \end{bmatrix}, \underline{b}_c = \begin{bmatrix} 0 \\ 0 \\ \dots \\ 0 \\ 1 \end{bmatrix}$$

Puisque A_c est une matrice stable, alors il existe une matrice P unique, définie positive et symétrique d'ordre $n \times n$ qui satisfait l'équation de Lyapunov suivante [25] :

$$A_c^T P + P A_c = -Q \quad 3.40$$

Où Q est une matrice définie positive arbitraire d'ordre $n \times n$. soit $V_e = \frac{1}{2} \underline{e}^T P \underline{e}$, alors en utilisant (3.35) et (3.37) nous obtenons

$$\dot{V}_e = -\frac{1}{2} \underline{e}^T Q \underline{e} + \underline{e}^T P \underline{b}_c ([\hat{f}(\underline{x}/\underline{\theta}_f) - f(\underline{x})] + [\hat{g}(\underline{x}/\underline{\theta}_g) - g(\underline{x})]u_c - d) \quad 3.41$$

Donc, nous devons avoir $\dot{V}_e \leq 0$ lorsque V_e dépasse une certaine constante \tilde{V} , mais de l'équation (3.41) nous remarquons qu'il est très difficile de concevoir u_c tel que le dernier terme de (3.37) soit négatif. Ce problème peut être résolu en ajoutant un autre terme de commande u_s à la commande u_c . Donc, la loi de commande finale devient alors :

$$u = u_c + u_s \quad 3.42$$

Ce terme de commande que nous avons ajouté appelé commande de supervision [25]. Le but de ce dernier est de forcer $\dot{V}_e \leq 0$ lorsque $V_e \geq \tilde{V}$. En substituant (3.37) dans (3.38) et en utilisant les mêmes manipulation utilisées dans l'obtention de (3.41), l'équation d'erreur devient :

$$\dot{\underline{e}} = A_c \underline{e} + \underline{b}([\hat{f}(\underline{x}/\underline{\theta}_f) - f(\underline{x})] + [\hat{g}(\underline{x}/\underline{\theta}_g) - g(\underline{x})]u_c - g(\underline{x})u_s - d) \quad 3.43$$

En utilisant (3.42) et (3.43), nous obtenons :

$$\begin{aligned} \dot{V}_e &= -\frac{1}{2}\underline{e}^T Q \underline{e} + \underline{e}^T P \underline{b}_c ([\hat{f}(\underline{x}/\underline{\theta}_f) - f(\underline{x})] + [\hat{g}(\underline{x}/\underline{\theta}_g) - g(\underline{x})]u_c - g(\underline{x})u_s - d) \\ &\leq -\frac{1}{2}\underline{e}^T Q \underline{e} + |\underline{e}^T P \underline{b}_c| (|[\hat{f}(\underline{x}/\underline{\theta}_f) - f(\underline{x})]| + |[\hat{g}(\underline{x}/\underline{\theta}_g)u_c - g(\underline{x})u_s]| + |d|) \\ &\quad - \underline{e}^T P \underline{b}_c g(\underline{x})u_s \end{aligned} \quad 3.44$$

Notre but maintenant est de concevoir u_s tel que $\dot{V}_e \leq 0$. Pour réaliser ça, nous aurons besoin d'utiliser l'hypothèse suivante :

Hypothèse

Supposons que nous pouvons déterminer des fonctions $f^U(\underline{x})$, $g^U(\underline{x})$ et $g_L(\underline{x})$ tel que $|f(\underline{x})| \leq f^U(\underline{x})$ et $g_L(\underline{x}) \leq g(\underline{x}) \leq g^U(\underline{x})$ pour $\underline{x} \in U_c$, ou $f^U(\underline{x}) < \infty$, $g^U(\underline{x}) < \infty$ et $g_L(\underline{x}) > 0$ pour $\underline{x} \in U_c$. En plus, supposons que la perturbation externe est bornée, c'est-à-dire, $|d| \leq d_m$, ou d_m est la borne supérieure de d .

En utilisant f^U , g^U et g_L , u_s devient alors

$$u_s = \begin{cases} \text{sgn}(\underline{e}^T P \underline{b}_c) \frac{1}{g_L(\underline{x})} [|\hat{f}| + f^U + |\hat{g}u_c| + |g^U u_c| + d_m] & \text{si } V_e \geq \tilde{V} \\ 0 & \text{si } V_e < \tilde{V} \end{cases} \quad 3.45$$

En substituant (3.44) dans (3.45), et en considérant le cas d'activation de u_s , nous obtenons :

$$\begin{aligned} \dot{V}_e &= -\frac{1}{2}\underline{e}^T Q \underline{e} + \left| \underline{e}^T P \underline{b}_c \right| (|\hat{f}| + |f| + |\hat{g}u_c| + |gu_c| + |d|) \\ &\quad - \frac{g}{g_L} (|\hat{f}| + f^U + |\hat{g}u_c| + |g^U u_c| + d_m) \\ &\leq -\frac{1}{2}\underline{e}^T Q \underline{e} \leq 0 \end{aligned} \quad 3.46$$

Par conséquent, l'utilisation de la commande de supervision u_s de (3.36), nous garantira toujours que $V_e < \tilde{V}$, puisque $P > 0$, alors V_e est bornée, ce qui implique que \underline{e} est aussi bornée, donc l'état \underline{x} est aussi borné.

Dans la suite, nous allons remplacer \hat{f} et \hat{g} par les systèmes flous type-2, et pour ajuster leurs paramètres $\underline{\theta}_f$ et $\underline{\theta}_g$, nous devons dériver les lois adaptatives.

L'estimation des paramètres optimaux $\underline{\theta}_f^*$ et $\underline{\theta}_g^*$ est donnée par

$$\underline{\theta}_f^* = \arg \min_{\underline{\theta}_f \in \Omega_f} [\sup_{\underline{x} \in \Omega_x} |\hat{f}(\underline{x}/\underline{\theta}_f) - f(\underline{x})|] \quad 3.47$$

$$\underline{\theta}_g^* = \arg \min_{\underline{\theta}_g \in \Omega_g} [\sup_{\underline{x} \in \Omega_x} |\hat{g}(\underline{x}/\underline{\theta}_g) - g(\underline{x})|] \quad 3.48$$

Où Ω_f , Ω_g et Ω_x sont des ensembles compacts convenables pour borner $\underline{\theta}_f$, $\underline{\theta}_g$ et \underline{x} , respectivement, qui sont définies par $\Omega_f = \{\underline{\theta}_f / |\underline{\theta}_f| / M_f\}$, $\Omega_g = \{\underline{\theta}_g / |\underline{\theta}_g| / M_g\}$ et $\Omega_x = \{\underline{x} / |\underline{x}| / M_x\}$, où M_f , M_g et M_x sont des constants positifs.

Définissons l'erreur d'approximation minimale suivante

$$w = \hat{f}(\underline{x}/\underline{\theta}_f^*) - f(\underline{x}) + (\hat{g}(\underline{x}/\underline{\theta}_g^*) - g(\underline{x}))u_c - d \quad 3.49$$

la dynamique de l'erreur (3.49) peut être réécrite par

$$\dot{\underline{e}} = A_c \underline{e} - \underline{b}_c g(\underline{x})u_s + \underline{b}_c \left[(\hat{f}(\underline{x}/\underline{\theta}_f) - \hat{f}(\underline{x}/\underline{\theta}_f^*)) + (\hat{g}(\underline{x}/\underline{\theta}_g) - \hat{g}(\underline{x}/\underline{\theta}_g^*))u_c + w \right] \quad 3.50$$

En substituant (3.47), (3.48) dans (3.50), l'équation précédente devient

$$\dot{\underline{e}} = A_c \underline{e} - \underline{b}_c g(\underline{x})u_s + \underline{b}_c w + \underline{b}_c \left[\underline{\phi}_f^T \underline{\xi}_f(\underline{x}) + \underline{\phi}_g^T \underline{\xi}_g(\underline{x})u_c \right] \quad 3.51$$

Ou

$\underline{\phi}_f = \underline{\theta}_f - \theta_f^*$ et $\underline{\phi}_g = \underline{\theta}_g - \theta_g^*$; $\underline{\xi}_f(\underline{x})$ et $\underline{\xi}_g(\underline{x})$ sont les FBFs moyennes de \hat{f} et \hat{g} , respectivement.

Considérannt maintenant la fonction de Lyapunov

$$V = \frac{1}{2} \underline{e}^T P \underline{e} + \frac{1}{2\gamma_1} \underline{\phi}_f^T \underline{\phi}_f + \frac{1}{2\gamma_2} \underline{\phi}_g^T \underline{\phi}_g \quad 3.52$$

Ou γ_1 et γ_2 sont des constantes positives. La dérivée temporelle de V est

$$\begin{aligned} \dot{V} = & -\frac{1}{2} \underline{e}^T Q \underline{e} - g(\underline{x}) \underline{e}^T P \underline{b}_c u_s + \underline{e}^T P \underline{b}_c w \\ & + \frac{1}{\gamma_1} \underline{\phi}_f^T \left[\dot{\underline{\theta}}_f + \gamma_1 \underline{e}^T P \underline{b}_c \underline{\xi}_f(\underline{x}) \right] + \frac{1}{\gamma_2} \underline{\phi}_g^T \left[\dot{\underline{\theta}}_g + \gamma_2 \underline{e}^T P \underline{b}_c \underline{\xi}_g(\underline{x}) u_c \right] \end{aligned} \quad 3.53$$

Selon (3.52) et $g(\underline{x}) \geq 0$, nous avons $g(\underline{x}) \underline{e}^T P \underline{b}_c u_s \geq 0$. Si nous choisissons les lois adaptatives suivantes :

$$\dot{\underline{\theta}}_f = -\gamma_1 \underline{e}^T P \underline{b}_c \underline{\xi}_f(\underline{x}) \quad 3.54$$

$$\dot{\underline{\theta}}_g = -\gamma_2 \underline{e}^T P \underline{b}_c \underline{\xi}_g(\underline{x}) u_c \quad 3.55$$

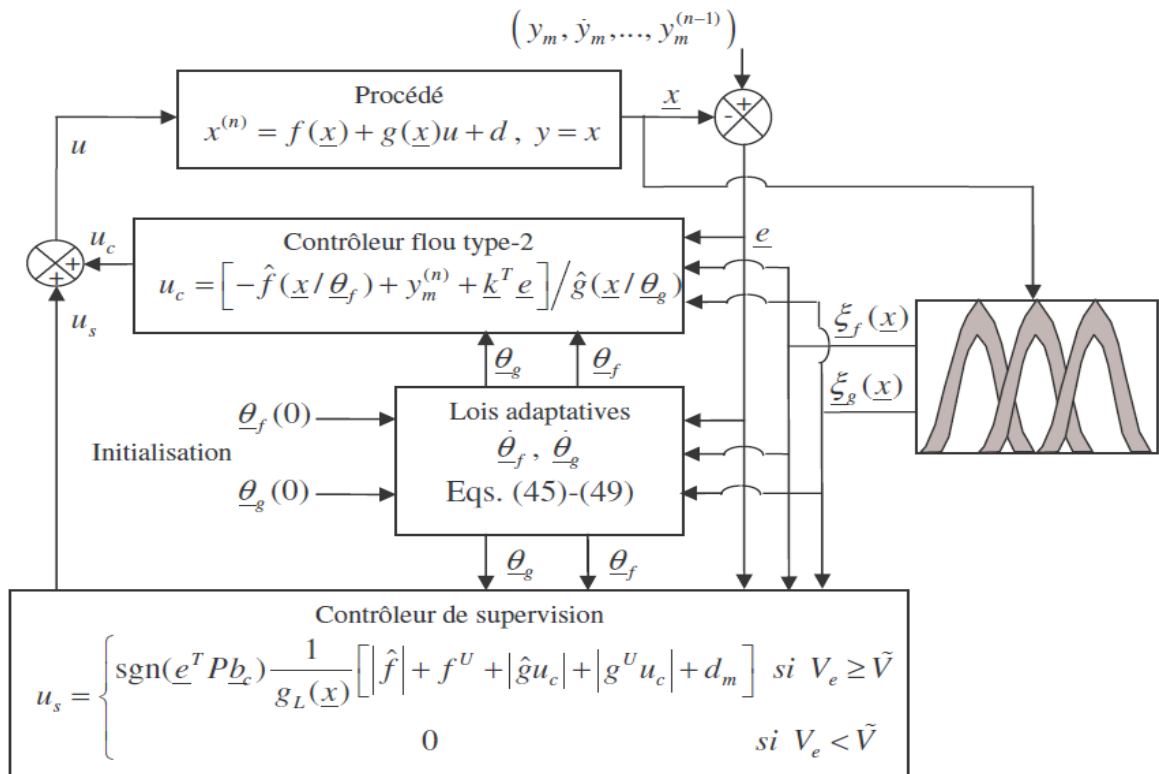


Figure. 3.9 Structure de Commande Adaptative Floue

Alors (3.53) devient

$$\dot{V} \leq -\frac{1}{2}\underline{e}^T Q \underline{e} + \underline{e}^T P b_c w \quad 3.56$$

Puisque le terme $\underline{e}^T P b_c w$ est de l'ordre de l'erreur d'approximation manimale, alors son influence est minimale. Si $w=0$, alors de (3.52) nous obtenons

$$\dot{V} \leq -\frac{1}{2}\underline{e}^T Q \underline{e} \leq 0 \quad 3.57$$

La lois adaptatives (3.54) et (3.55) ne peuvent pas garantir que $\theta_f \in \Omega_f$ et $\theta_g \in \Omega_g$. Par conséquent, les lois adaptatives doivent être modifiées et cela en utilisant l'algorithme de projection des paramètres [25], tel que les vecteurs paramètres restent à l'intérieur des domaines contraints. Les lois modifiées sont données pas

- On utilise la loi adaptative suivante pour ajuster le vecteur paramètre $\underline{\theta}_f$:

$$\dot{\underline{\theta}}_f = \begin{cases} -\gamma_1 \underline{e}^T P \underline{b}_c \underline{\xi}_f(\underline{x}) & \text{si } (|\theta_f| < M_f \text{ ou } |\theta_f| = M_f \text{ et } \underline{e}^T P \underline{b}_c \underline{\theta}_f^T \underline{\xi}_f(\underline{x}) \geq 0) \\ \text{proj} \{ -\gamma_1 \underline{e}^T P \underline{b}_c \underline{\xi}_f(\underline{x}) \} & \text{si } (|\theta_f| = M_f \text{ et } \underline{e}^T P \underline{b}_c \underline{\theta}_f^T \underline{\xi}_f(\underline{x}) < 0) \end{cases} \quad 3.58$$

Où l'opérateur de projection $\text{proj}\{*\}$ est défini par

$$\text{proj} \{ -\gamma_1 \underline{e}^T P \underline{b}_c \underline{\xi}_f(\underline{x}) \} = -\gamma_1 \underline{e}^T P \underline{b}_c \underline{\xi}_f(\underline{x}) + \gamma_1 \underline{e}^T P \underline{b}_c \frac{\underline{\theta}_f \underline{\theta}_f^T}{|\underline{\theta}_f|^2} \underline{\xi}_f(\underline{x}) \quad 3.59$$

- On utilise la loi adaptative suivante pour ajuster le vecteur paramètre $\underline{\theta}_g$:

- Quand un élément θ_g^i de $\underline{\theta}_g = \epsilon$, utiliser

$$\theta_g^i = \begin{cases} -\gamma_2 \underline{e}^T P \underline{b}_c \xi_g^i(\underline{x}) u_c & \text{si } \underline{e}^T P \underline{b}_c \xi_g^i(\underline{x}) u_c < 0 \\ 0 & \text{sinon} \end{cases} \quad 3.60$$

Où $\xi_g^i(\underline{x})$ est la i ème composante de $\underline{\xi}_g(\underline{x})$.

- Autrement, on utilise

$$\dot{\underline{\theta}}_g = \begin{cases} -\gamma_2 \underline{e}^T P \underline{b}_c \underline{\xi}_g(\underline{x}) u_c & \text{si } (|\theta_g| < M_g \text{ ou } |\theta_g| = M_g \text{ et } \underline{e}^T P \underline{b}_c \underline{\theta}_g^T \underline{\xi}_g(\underline{x}) u_c \geq 0) \\ \text{proj} \{ -\gamma_2 \underline{e}^T P \underline{b}_c \underline{\xi}_g(\underline{x}) u_c \} & \text{si } (|\theta_g| = M_g \text{ et } \underline{e}^T P \underline{b}_c \underline{\theta}_g^T \underline{\xi}_g(\underline{x}) u_c < 0) \end{cases} \quad 3.61$$

Ou l'opérateur de projection $proj\{*\}$ est défini par :

$$proj \left\{ -\gamma_2 \underline{e}^T P \underline{b}_c \underline{\xi}_g(\underline{x}) u_c \right\} = -\gamma_2 \underline{e}^T P \underline{b}_c \underline{\xi}_g(\underline{x}) u_c + \gamma_2 \underline{e}^T P \underline{b}_c \frac{\underline{\theta}_g \underline{\theta}_g^T}{|\underline{\theta}_g|^2} \underline{\xi}_g(\underline{x}) u_c \quad 3.62$$

3.4.3. Application au Modèle inverse flou

Pour tirer plus d'avantage de notre modèle inverse flou, la commande adaptative floue proposée a été appliquée à la commande en vitesse de la machine asynchrone en utilisant notre modèle inverse flou.

D'après le modèle de la machine asynchrone présenter en (2.6), il est clair que $0 < g < \infty$, ce qui assure l'hypothèse citer précédemment, donc il nous reste à déterminer les bornes f^U, g^U et g_L par :

$$|f(x)| \leq 766 \Rightarrow f^U \equiv 766 \quad , \quad |g(x)| \leq 62 \Rightarrow g^U \equiv 62 \quad \text{et} \quad |g(x)| \geq 0.008 \Rightarrow g_L \equiv 0.008$$

Notre objectif est de contrôler l'état x_5 du système pour qu'il suive la trajectoire de référence ou bien la consigne. La perturbation externe d est le couple résistant.

En suivant la procédure de l'algorithme proposé, la conception du régulateur sera assurée par les étapes suivantes [25]:

Etape 1 : il n y a pas de formule précise pour calculer le vecteur du gain de commande, donc il a été choisi par tâtonnement comme $k_1 = 1, k_2 = 1, k_3 = 1, k_4 = 1$ et $k_5 = 1$

Etape 2 : nous choisissons Q comme $= diag(5,5)$, alors, par résolution de (3.56), la matrice

symétrique P d'ordre 5×5 sera
$$\begin{bmatrix} 42 & 3 & 8 & 71 & 1 \\ 09 & 24 & 12 & 8 & 76 \\ 23 & 2 & 2 & 5 & 14 \\ 09 & 54 & 11 & 2 & 19 \\ 84 & 09 & 10 & 23 & 1 \end{bmatrix}$$
. P est défini positive.

Etape 3 : si nous exigeons que $|\Omega| = |x_5| \leq \Omega_n$ et $|C_{em}| = |u| \leq C_{em_n}$ alors nous pouvons sélectionner $M_f = 67, M_g = 76, \varepsilon = 0.98$ et $\tilde{V} = 0.89$.

Etape 4 :

- Pour calculer la loi de commande floue u , nous définissons trois ensembles flous intervalles sur chacun des axes des cinq variables $x_i, i = 1:5$; notés $\tilde{F}_i^1, \tilde{F}_i^2$ et \tilde{F}_i^3 . nous devons noter que dans [25 et 61] les ensembles flous ont été choisies à cinq. Des sigmoïdes ont été choisis comme des fonctions d'appartenance.
- Pour couvrir tous les cas, nous appliquons $M = m_1 \times m_2 \times m_3 \times m_4 \times m_5 = 3 \times 3 \times 3 \times 3 \times 3$ règles floues pour chaque'un de $\hat{f}(x/\theta_f)$ et $\hat{g}(x/\theta_g)$. nous devons aussi noter que dans [138,139 et 142], les auteurs ont utilisé mois de règles floue.
- Calculer les fonctions de base floue $\underline{\xi}_f(x)$ et $\underline{\xi}_g(x)$ par conséquent, $\hat{f}(x/\theta_f)$ et $\hat{g}(x/\theta_g)$ seront obtenues.

Etape 5 : Construire l'action de commande en utilisant (3.42), puis l'appliquer au procédé. Calculer les lois de commande adaptatives pour ajuster les vecteurs paramétriques θ_f et θ_g . Les valeurs initiales θ_f et θ_g sont choisies aléatoirement.

La structure complète de schéma de commande ainsi proposé est le suivante figure 3.10.

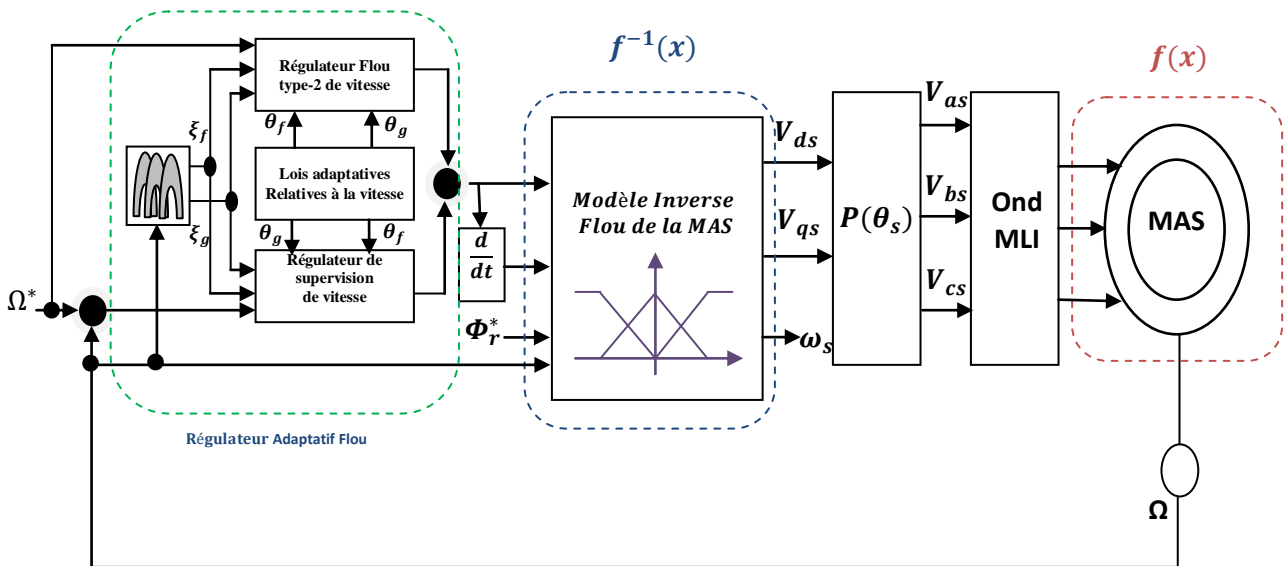


Figure 3.10 Structure de commande adaptative floue bas é sur le modèle inverse flu proposée pour la commande du moteur à induction

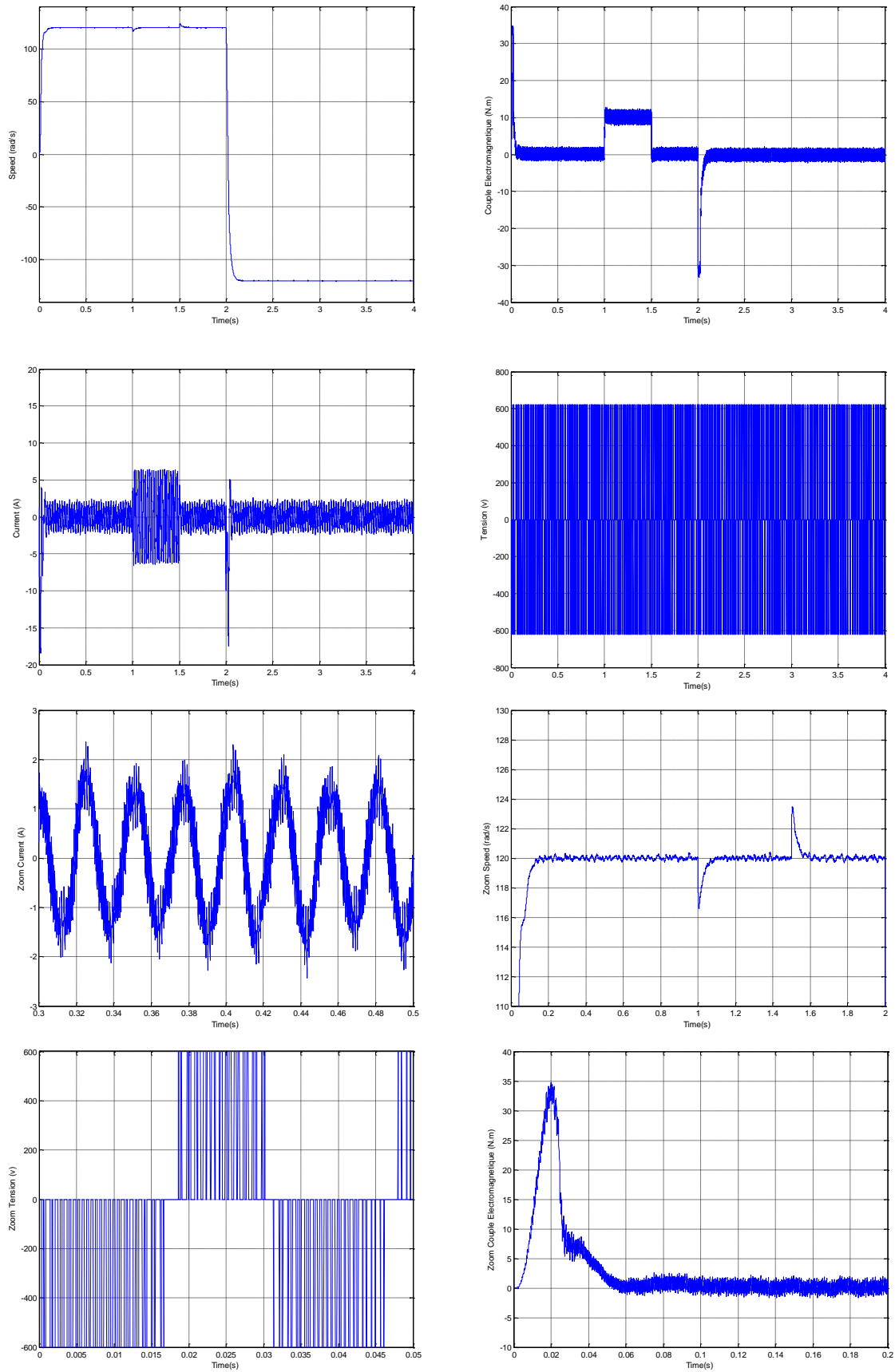


Figure 3.11 Les résultats de simulation de la commande adaptative Floue basé sur le modèle inverse flou de la MAS

Une simulation numérique a été faite pour prouver l'efficacité de cette technique, il est clair que les résultats obtenus sont nettement meilleurs. Ainsi, nous pouvons dire que la logique floue a permis de prouver son efficacité encore une fois en l'appliquant sur un régulateur adaptative classique. Dans cette partie, nous avons présenté une approche de commande adaptative floue pour commander le moteur à induction sur la base du modèle inverse flou. Ce contrôleur a été basé sur la logique floue type-2. Nous avons également proposé une méthode de calcul pour déterminer les FBFs du régulateur proposé. Nous avons montré comment le régulateur flou type-2 a diminué le nombre de règles par rapport aux travaux déjà existants dans la littérature, ce qui permet bien sûr de diminuer la complexité du système. Nous avons également constaté que l'effort de commande pour ce type de régulateurs a été réduit. Les résultats de simulations ont montré l'efficacité de notre approche.

3.5 Conclusion

L'utilisation d'un régulateur IP à la commande de la machine asynchrone basée sur son modèle inverse flou a permis de résoudre le problème blocage par saturation du régulateur PID classique. Dans ce contexte, les propriétés offertes par ce dernier ont été mises à profit pour développer un régulateur permet de surmonter le blocage par saturation afin de générer une loi de commande fiable sur une large plage de variation de la variable commander.

Nous avons proposé d'utiliser la logique floue, ou bien de modéliser le régulateur par mode glissant classique en utilisant la technique intelligente par la logique floue, afin d'obtenir un régulateur par mode glissant flou, robuste fiable.

Et finalement, et toujours afin de tirer profit du modèle inverse flou, une loi de commande adaptative floue a été appliquée sur le modèle, et d'après les résultats de simulation, nous pouvons dire que la commande robuste floue peut s'avérer très intéressante surtout pour les régulateurs difficilement implémentables.

Chapitre 4

Chapitre 4

Dispositif Expérimental

4.1. Introduction

Nous allons dans ce chapitre détailler la mise en œuvre expérimentale des stratégies de commande dont la faisabilité a été testée en simulation fine, dite quasi expérimentation dans l'environnement MATLAB. Dans une première partie, nous détaillons les éléments de puissance et leur connexion. Puis, nous abordons dans une deuxième partie les aspects de l'informatique industrielle avec la commande numérique par une carte dSPACE à base d'un processeur DSP, les signaux et leur mode de communication. Nous verrons également l'implémentation sous le logiciel ControlDesk. Les résultats expérimentaux ont été donnés avec ceux de la simulation durant les chapitres précédents.

4.2. Identification des paramètres de la MAS

Il existe de nombreuses méthodes d'identification des paramètres de la machine asynchrone, différentes de par leur principe et leur complexité. Les méthodes classiques sont basées sur des essais en régime permanent sinusoïdal [9 et 42]. D'autres méthodes tiennent compte du comportement transitoire électrique et mécanique de la machine et qui consistent à traiter les données issues des relevés de vitesse, tension et courant suite à un démarrage à vide sous pleine tension. Et ce en utilisant des méthodes qui minimisent l'écart quadratique entre les mesures et les résultats des modèles proposés [9]. Il existe encore d'autres méthodes basées sur l'analyse du comportement de la machine avec une alimentation continue [9].

Le travail d'identification est un travail à part entière, souvent associé à l'aspect modélisation et commande de la machine. Il est nécessaire de connaître les paramètres de la machine pour la commander. Ainsi, pour la validation des modèles proposés, nous avons adopté les méthodes classiques avec alimentation sinusoïdale. Ces méthodes classiques sont largement suffisantes pour l'identification des paramètres avec une précision acceptable.

Pour la machine asynchrone à cage il est impossible d'accéder à des informations directes issues de mesures faites au rotor, il faut donc aboutir à ces grandeurs rotorique à partir des grandeurs mesurables. Les méthodes classiques sont basées sur l'essai à vide et l'essai à rotor bloqué. Ces essais ne suffisent pas pour déterminer séparément tous les paramètres et souvent on a recours à des hypothèses supplémentaires. Parmi ces hypothèses celle qui consiste à prendre les inductances de fuite statorique est rotorique égales. Ce qui signifie que la machine a été construit pour avoir une induction uniforme dans le circuit magnétique.

Les paramètres nécessaires à la simulation du fonctionnement de la machine se divisent en paramètres électromagnétiques et mécaniques. Nous montrons plus loin que les paramètres électromagnétiques requis sont (R_s , R_r , L_s , L_r et M).

Les essais sont effectués sur une machine asynchrone de 1.5 KW. Nous avons retenu les méthodes qui tiennent compte du comportement transitoire aussi bien électrique que mécanique de la machine.

Comme les essais expérimentaux seront effectués en régime sinusoïdal, nous faisons appel au modèle de la Fig.2.1 représentant le schéma monophasé équivalent ramené au stator. Dans ce schéma, N_r et R_r représentent respectivement l'inductance de fuite totale et la résistasse rotorique ramenées au stator. L'expression de N_r est donnée par [48] :

$$N_r = \frac{L_s L_r}{M} - M \quad 4.1$$

Nous avons fait les essais sur une machine de puissance: 1.5 kW.

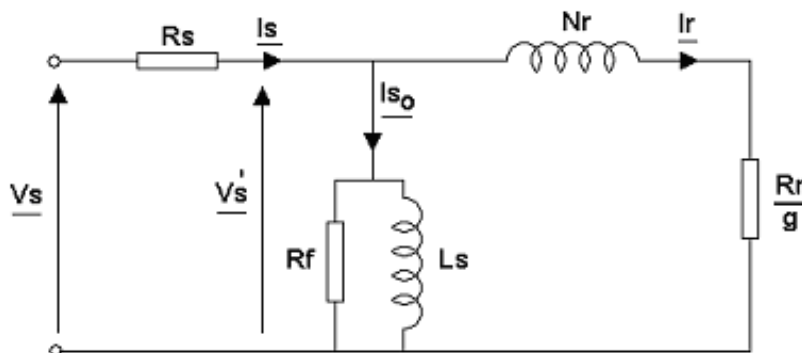


Figure.4.1. Schéma monophasé équivalent ramené au stator

4.2.1. Mesure de la résistance statorique

La mesure de la résistance statorique est effectuée en alimentant la machine en continu. Vu l'ordre de grandeur des résistances il est difficile d'appliquer des tensions élevées ainsi il est préférable d'utiliser le montage étoile et alimenter deux phases en série Figure.4.2.

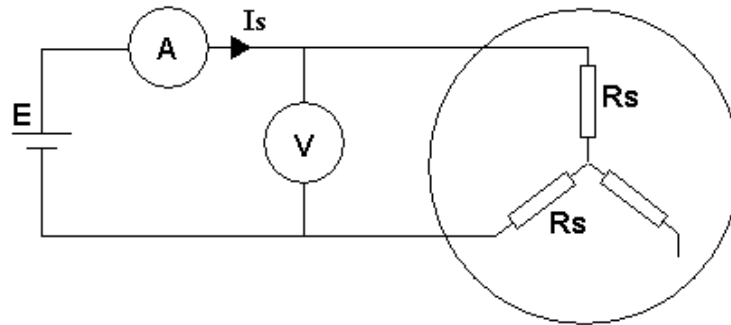


Figure.4.2. Méthode volt-ampérométrique

On a réalisé deux mesures pour différentes positions de l'arbre. La valeur de la résistance est obtenue simplement par la relation :

$$R_s = \frac{E}{2I_s} = 5.62 \Omega \quad 4.2$$

Les valeurs obtenues sont souvent entachées d'erreurs dues essentiellement à la température des enroulements, aux chutes de tension dans les connections et les fils de liaison.

4.2.2. Essai à vide

Le tableau suivant regroupe une série de mesures effectuées pour différentes valeurs de la tension U (Tableau.4.1).

Tension U (V)	Iso (A)	Pso (W)	Qso (VAR)	N (tr/min)	Pfs+Pméca (W)	U ² (V ²)
380	1.33	220	946	1497	190.62	144400
369	1.27	204	870	1497	176.81	136161
350	1.18	184	766	1496	160.12	122500
280	0.89	152	464	1495	139.67	78736.37
250	0.80	140	411	1494	132.88	72521.5

Tableau.4.1. Tableau des mesures effectuées pour différentes valeurs de la tension U

Dans un fonctionnement à vide, le moteur asynchrone absorbe une puissance P_{so} qui correspond à la somme des pertes mécaniques, ferromagnétiques (fer) et Joules au stator.

$$P_{so} = P_{js} + P_{fs} + P_{méca} \quad 4.3$$

Où les pertes mécaniques sont la somme de pertes par frottements (proportionnelles à la vitesse) et des pertes par ventilation (proportionnelles au carré de la vitesse) les pertes fer sont les pertes par courants de Foucault et par Hystérésis (proportionnelles au carré de la tension V_s , à l'induction B , la fréquence). Les pertes Joules statoriques (3phases) s'expriment par la relation :

$$P_{js} = 3R_s I_s^2 \quad 4.4$$

Puisque l'essai est réalisé à vide, on peut considérer $I_r=0$ d'où $P_{jr} = P_{fr} = 0$

En traçant la courbe de la puissance $P_{fs} + P_{méca}$ en fonction du carré de la tension d'alimentation, on obtient en principe une droite dont le prolongement jusqu'à la tension nulle donne les pertes mécanique, soit :

$$P_{méca} = 72.93W$$

En négligeant la chute de tension aux bornes de R_s , les pertes fer peuvent s'exprimer sous la forme:

$$P_f = \frac{3}{R_f} V_s^2 \quad 4.5$$

L'inductance cyclique L_s est donnée par la relation :

$$L_s = 3 \frac{V_s^2}{Q_{so} \omega_s} \quad 4.6$$

Application numérique :

$$R_f = 1970 \Omega$$

$$L_s = L_r = 0.48 H$$

4.2.3. Détermination des paramètres mécaniques

La détermination du coefficient de frottement visqueux et du moment d'inertie est basée sur la mesure des pertes mécanique lorsque la machine tourne à une vitesse donnée et sur le relevé de la courbe de ralentissement. D'après la Figure.4.3, le moment d'inertie J peut être calculée par :

$$J = \frac{P_{\text{mec}}}{\Omega_n \left(\frac{d\Omega}{dt} \right)_{\Omega=\Omega_n}} \quad 2.7$$

Cette relation montre que la mesure du moment d'inertie J dépend de la précision avec laquelle la puissance mécanique est déterminée.

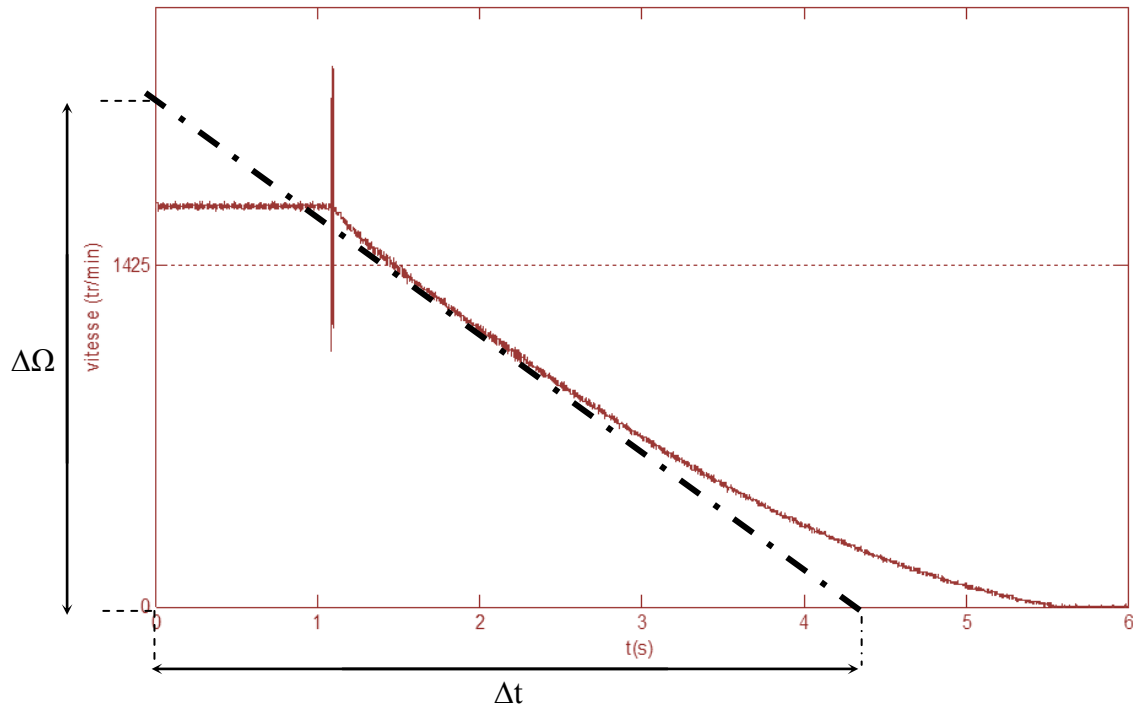


Figure.4.3. Essai de ralentissement

En régime permanent à vide, il est légitime de considérer que le couple résistant n'est dû qu'aux frottements visqueux ce qui permet de déduire le coefficient de frottement visqueux f à partir de la relation suivante [48] :

$$C_{\text{em}} = f \Omega_n \quad 4.8$$

On trouve :

$$J = 0.0049 \text{ Kg.m}^2$$

$$f = 0.0003 \text{ N.m.s/rd}$$

4.2.4. Essai en court-circuit

Le rotor étant bloqué ($g=1$), la tension statorique doit être ajustée de telle sorte que le courant statorique soit proche du courant nominal. Dans ces conditions, la tension appliquée

devient nettement faible devant la tension nominale, ce qui permet de négliger le courant I_{so} et réduire le schéma de la Fig.2.4 à ce qui suit:

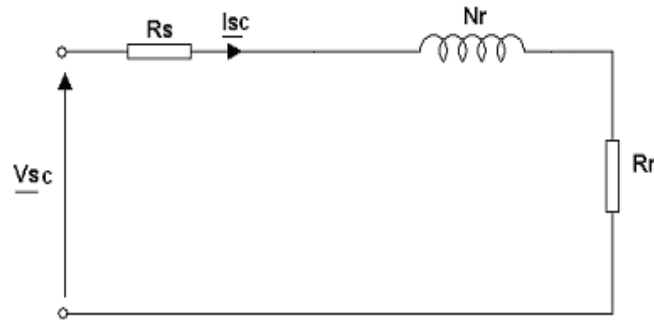


Figure.4.4. Essai avec rotor bloqué

Pour cet essai en court-circuit, on a relevé :

$$U_s=99 \text{ V}$$

$$I_s=3.2 \text{ A}$$

$$P_{sc}=319 \text{ W}$$

En négligeant les puissances dissipées dans R_f et L_s

$$P_{sc} \cong 3(R_s + R_r)I_{sc}^2 \quad 4.9$$

$$Q_{sc} \cong 3(N_r \omega_s)I_{sc}^2 \quad 4.10$$

On en déduit :

$$R_r = \frac{P_{sc}}{3I_{sc}^2} - R_s \quad 4.11$$

Cette résistance joue un rôle très important dans la mesure où elle a une influence sur la valeur du couple de démarrage et sur son allure.

De la même façon, le calcul de N_r découle de la relation 4.1 :

$$N_r = \frac{Q_{sc}}{3\omega_s I_{sc}^2} \quad 4.12$$

D'où :

$$R_r=4.26\Omega \text{ et } N_r=0.039\text{H}$$

4.2.5 Validation du modèle

La simulation numérique du fonctionnement des systèmes physiques est devenue une étape classique dans l'étude de ces systèmes. Elle consiste à résoudre le système d'équations donné par le modèle, afin d'obtenir les différentes grandeurs du système physique. Le choix du pas de simulation est très important, d'une part, pour bien suivre le comportement dynamique des variables rapides du système et, d'autre part, pour conserver un temps de simulation raisonnable par rapport au temps réel nécessaire pour le calcul.

Afin de comparer les modèles développés, nous avons procédé à une simulation d'un démarrage direct à vide de la machine asynchrone. Comme le système d'équations différentielles est non linéaire, nous avons choisi pour le résoudre la méthode explicite de Runge-Kutta d'ordre 4. Dans le but de valider nos modèles, nous avons monté un banc de mesures. Nous avons effectué un démarrage direct alimenté par un l'onduleur MLI pour un indice de modulation $m=17$ et nous avons mesuré les grandeurs suivantes : le courant statorique et la vitesse de rotation du moteur, nous avons utilisé des capteurs pour le courant. La vitesse a été déterminée grâce à un tachymètre. Les mesures recueillies sont ensuite envoyées à une carte DSP1103.

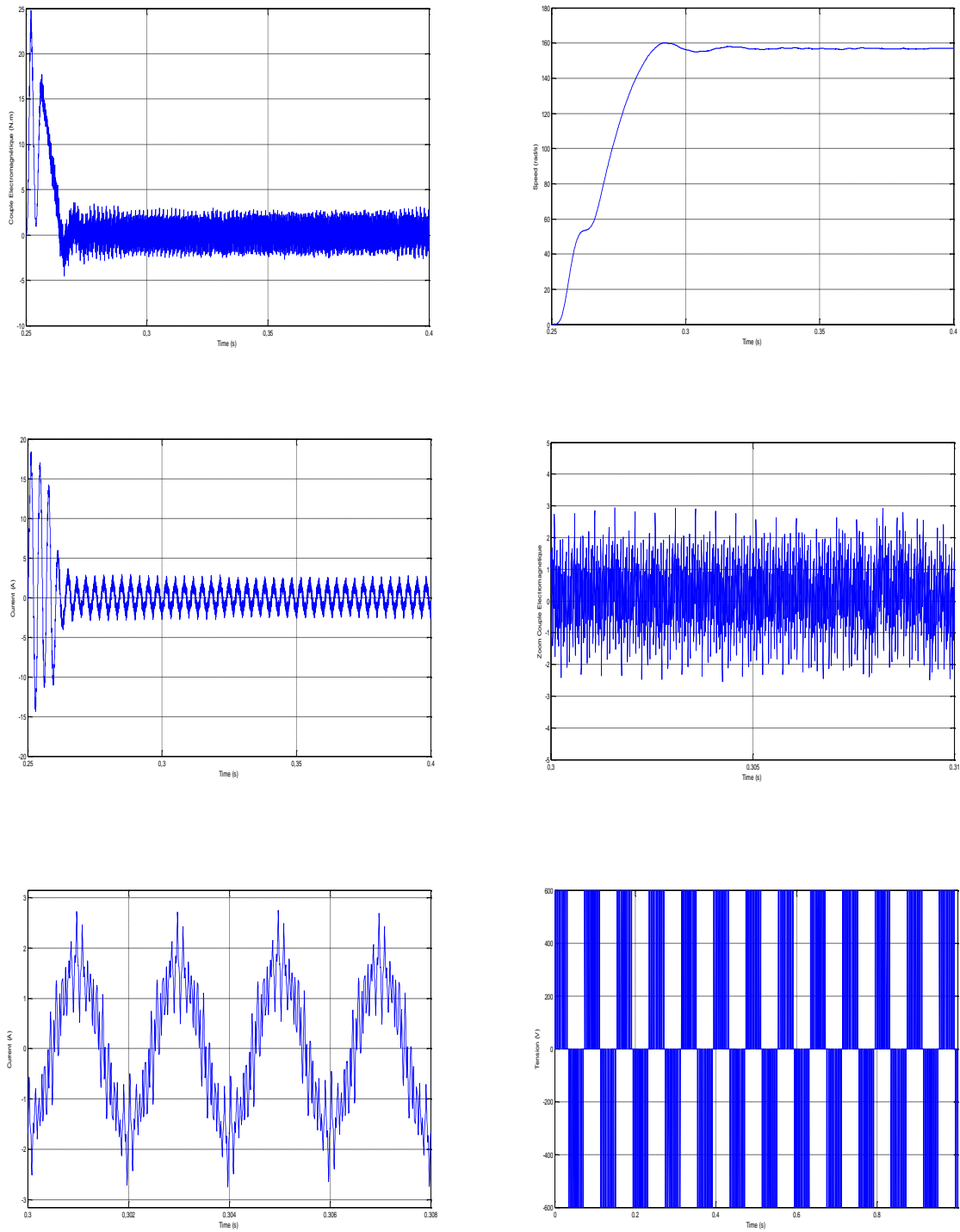


Figure.4.5 : Validation du modèle de la MAS - Simulation Numérique

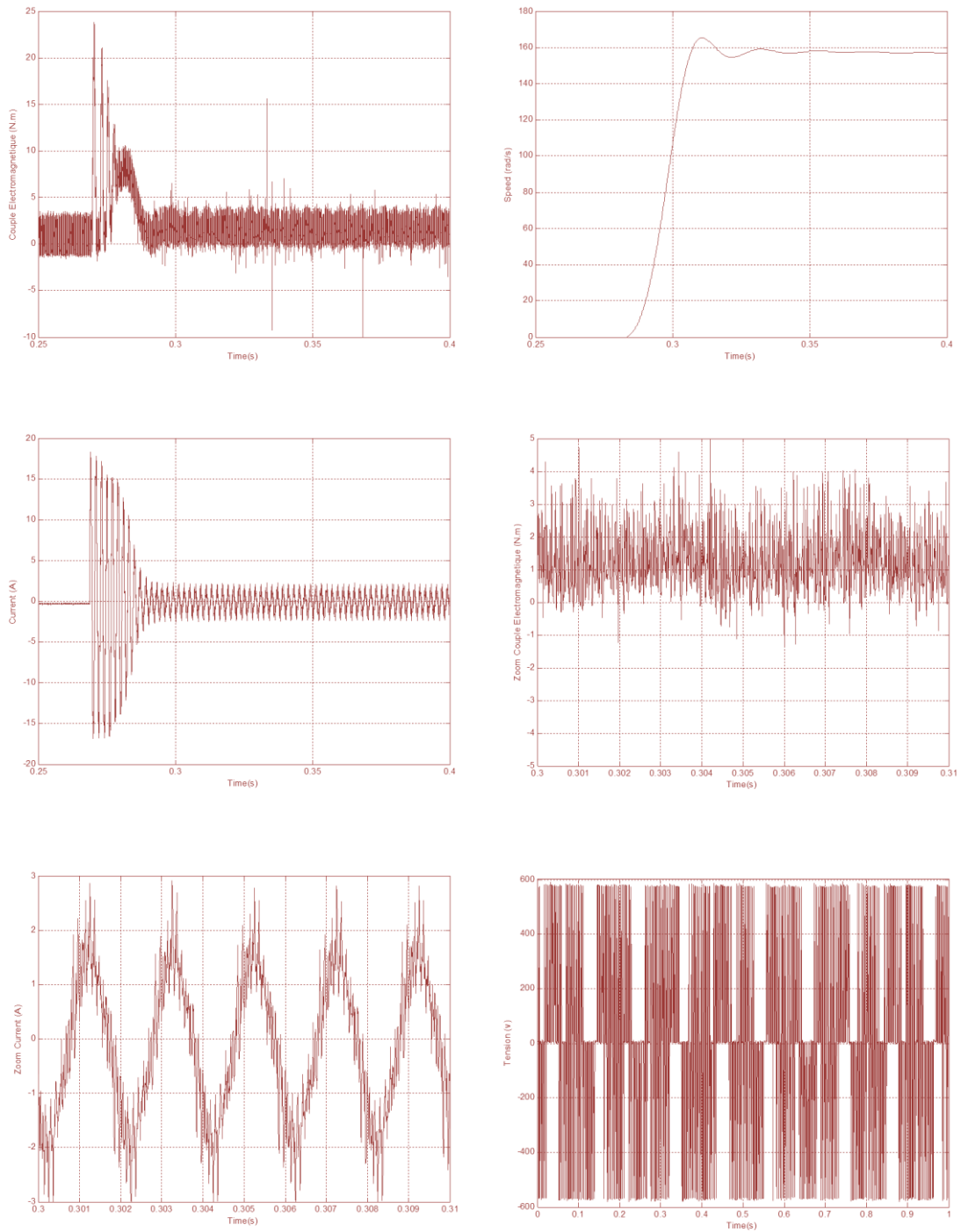


Figure.4.6. Validation du modèle de la MAS - Expérimentation

Par rapport aux travaux déjà effectués au sein de notre laboratoire et qui ont surtout utilisé les différentes méthodes, nous avons exploré la voie qu'offrent les essais classiques à l'identification des paramètres de la machine asynchrone. Ce sont des méthodes simple et rapides, elles ont l'avantage d'être beaucoup moins sensibles au point de départ et aux autres contraintes.

4.3. Description générale

4.3.1. Partie Puissance

4.3.1.1. Prototype MAS rotatif

Le prototype de machine rotative a induction destinée à diverse applications industrielles conçues par le puissant **Leroy Somer** [65]. Cette machine, grâce à l'accroissement de sa fréquence de fonctionnement, bénéficie comme toute machine asynchrone d'une réduction de masse et de volume et d'augmentation logique de ses performances massiques. Elle met en œuvre, comme nous l'avons dit, la technique du couplage par induction. L'architecture de base pour cette machine est représentée sur la figure 4.7.

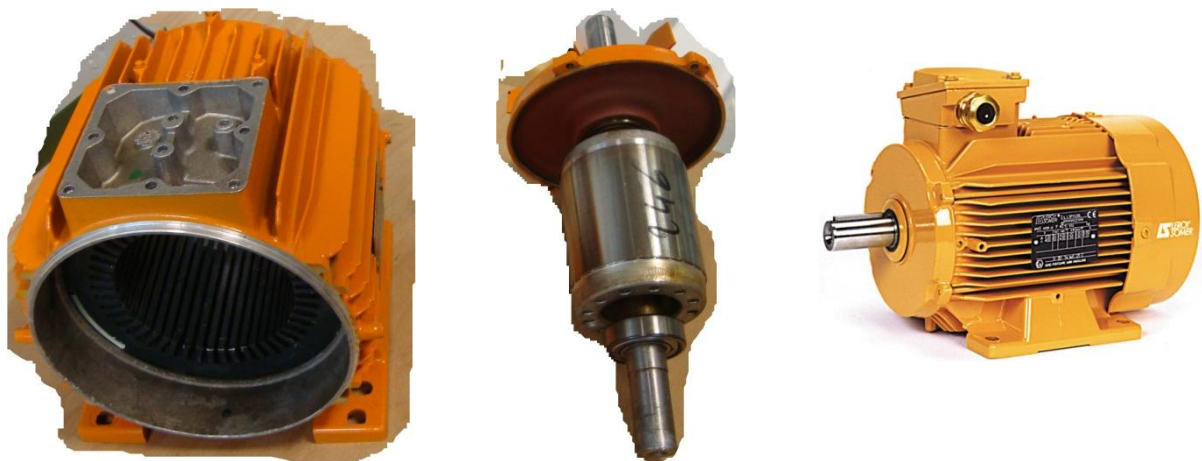
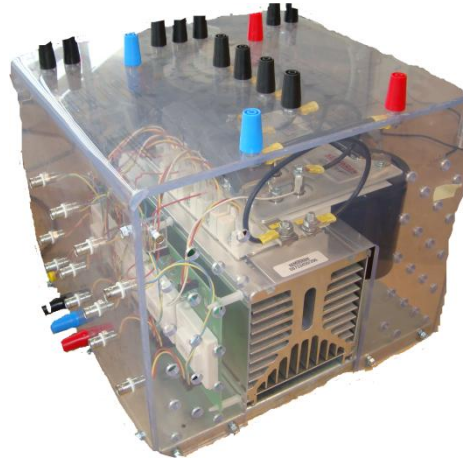


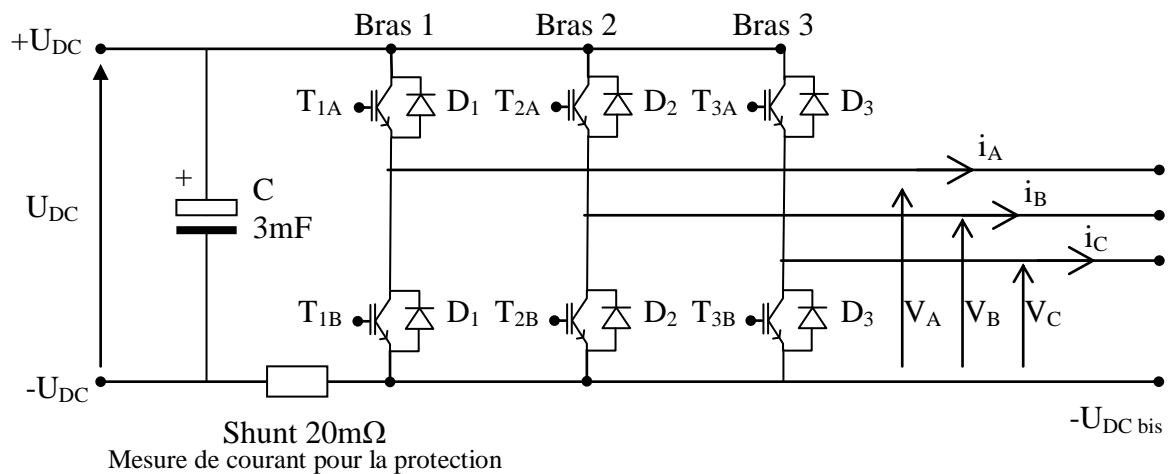
Figure.4.7. Prototype de la MAS

4.3.1.2. Prototype Convertisseur

Il est constitué d'un redresseur de tension à base de diodes de puissance (1000V/25A) et d'un onduleur de tension composé de six transistors IGBT (1000V/25A) qui sont commutés à une fréquence de 10 kHz. La carte d'interface permet d'attaquer les IGBT avec une MLI centrée générée à partir des six signaux MLI issus des comparateurs du DSP. Chaque IGBT est équipé d'une diode de roue libre (1000V/20A) mise en antiparallèle.



a)



b)

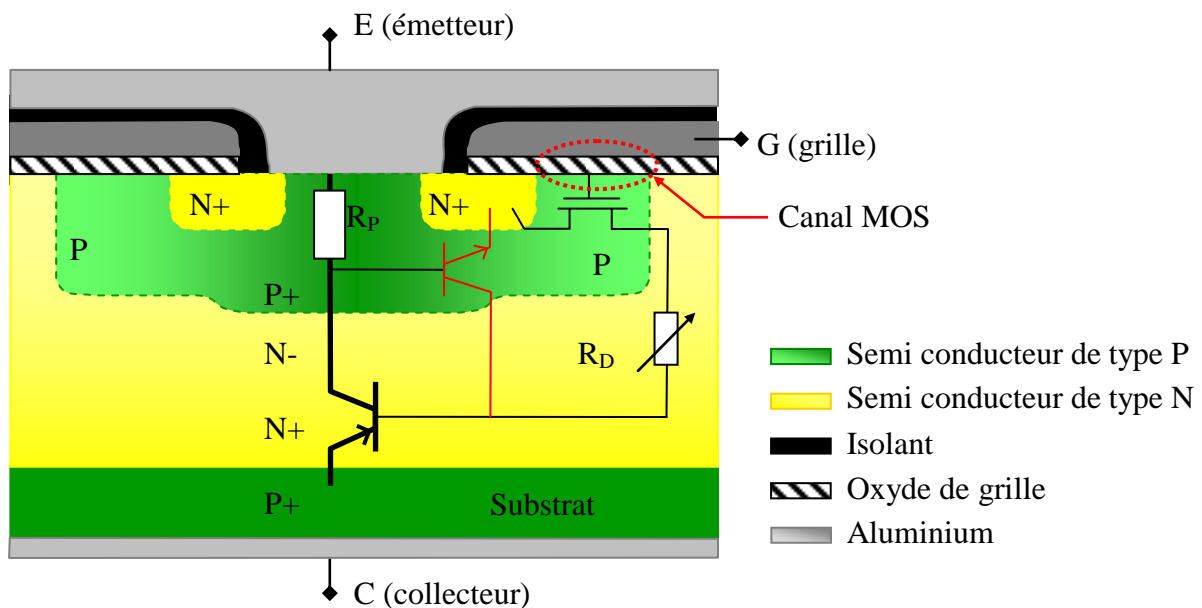
Figure 4.8 : Onduleur triphasé IES-GEM utilisé lors des tests expérimentaux;

a) Vue de dessus, b) schéma électrique.

4.3.1.2.1 Description de l'IGBT

Le transistor bipolaire à grille isolée (IGBT, de l'anglais Insulated Gate Bipolar Transistor) est un dispositif semi-conducteur de la famille des transistors qui est utilisé comme interrupteur électronique, principalement dans les montages de l'électronique de puissance. Ce composant, qui combine les avantages des technologies précédentes, c'est-à-dire la grande simplicité de commande du transistor à effet de champ par rapport au transistor bipolaire, tout en conservant les faibles pertes par conduction de ce dernier, a permis de nombreux progrès dans les applications de l'électronique de puissance, aussi bien en ce qui concerne la fiabilité que sur l'aspect économique[64]. Les transistors IGBT ont permis d'envisager des développements jusqu'alors non viables en particulier dans la vitesse variable ainsi que dans les applications des machines électriques et des convertisseurs de puissance qui nous accompagnent chaque jour et partout, sans que nous en soyons particulièrement conscients : automobiles, trains, métros, bus, avions, bateaux, ascenseurs, électroménager, télévision, domotique, etc.

Depuis la fin des années 90, l'IGBT s'est imposé comme l'interrupteur de puissance par excellence. Les études sur les IGBT sont nombreuses dans la littérature scientifique [64]. L'IGBT adopte deux formes complémentaires possibles, selon la nature N ou P de son canal MOS (Metal Oxide Semi conductor). L'IGBT à canal N est plus utilisé que l'IGBT à canal P à cause de ses performances électriques supérieures (pertes électriques, etc.)[64].



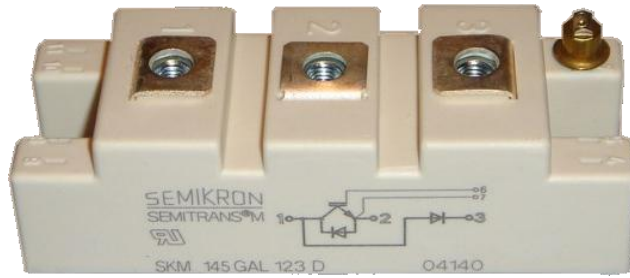


Figure 4.9 : module d'IGBT (cellule de commutation) de la marque SEMIKRON modèle SKM145GAL123D.

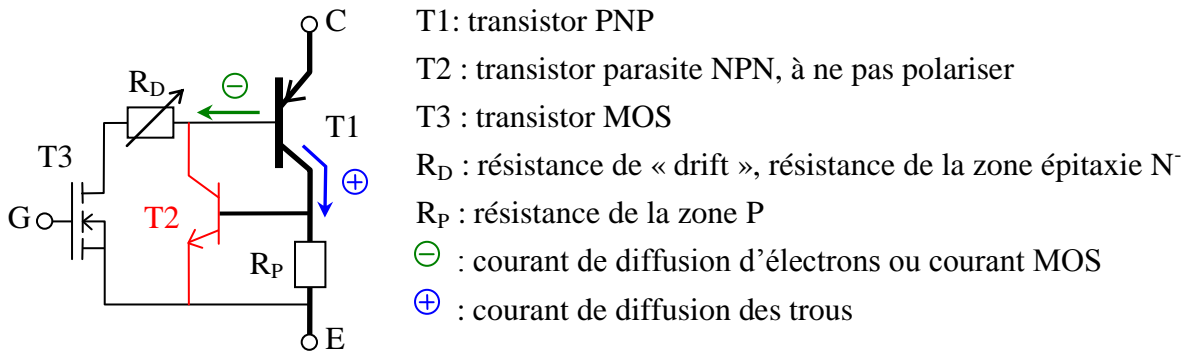


Figure 4.10 circuit équivalent naturel d'un IGBT à canal N.

La figure 4.8 présente la coupe transversale et le circuit équivalent d'une cellule IGBT à canal N. Pour un fonctionnement normal de l'IGBT (figure 4.9), le transistor bipolaire T2 NPN (ou le thyristor formé par T1 et T2) ne doit pas fonctionner pour éviter l'effet « latch up ». Le non fonctionnement de T2 permet d'obtenir un circuit équivalent simplifié de l'IGBT (figure 4.10).

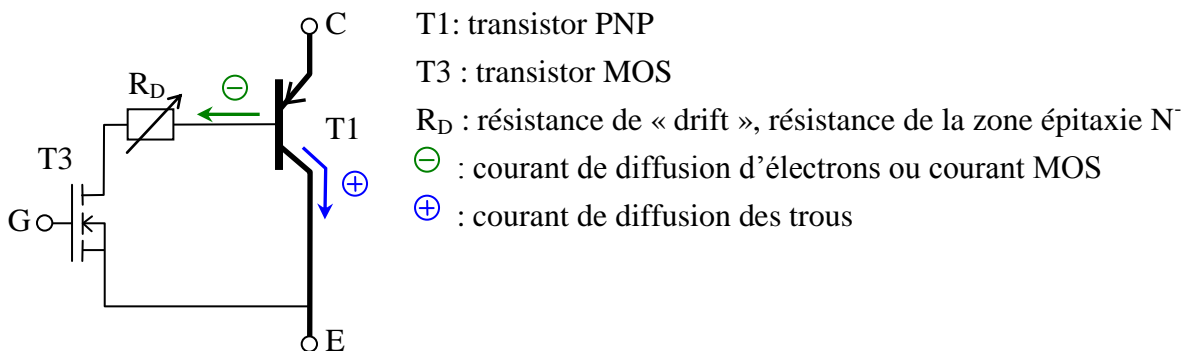


Figure 4.11 circuit équivalent simplifié d'un IGBT à canal N.

- **A l'état passant de l'IGBT**

L'état passant de l'IGBT est obtenu par polarisation adéquate de la grille de son transistor MOS T3 (figure 3.12) [64]. La fermeture de T3 entraîne la polarisation de T1, qui se met à conduire à son tour. Généralement, on représente la fermeture d'IGBT par son réseau de caractéristiques de courant et de tension (figure 3.12):

$$I_C = f(V_{CE})_{V_{GE}} \quad 4.13$$

Avec I_C le courant collecteur,

V_{CE} la tension directe de l'IGBT ou tension collecteur-émetteur et

V_{GE} la tension de polarisation ou tension grille-émetteur.

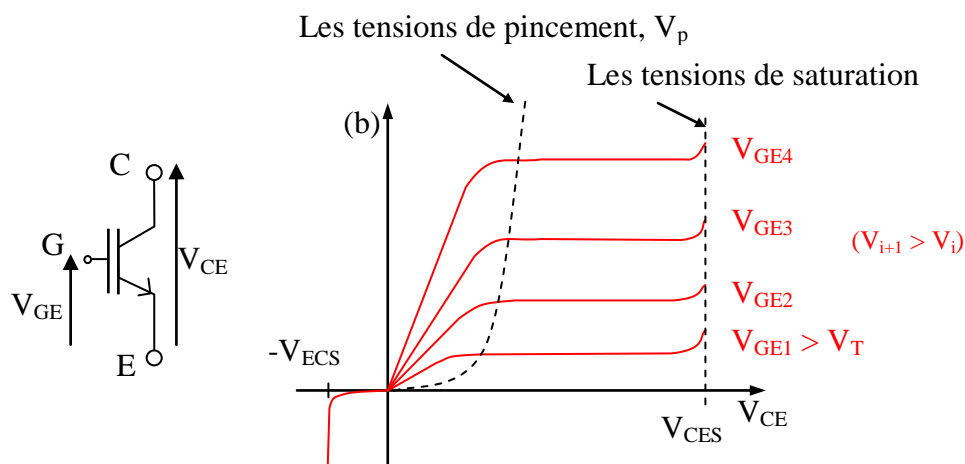


Figure 1.12 : symbole d'un IGBT à canal N et ses caractéristiques de sortie $I_C = f(V_{CE}, V_{GE})$ pour $V_{GE} > V_T$ (tension de seuil de l'IGBT).

Une caractéristique courant tension de l'IGBT est composée de :

Une phase ohmique. La tension V_{CE} est inférieure à la tension de pincement du canal MOS V_p . Le courant I_C varie linéairement avec V_{CE} . L'expression rapprochée du courant est (figure 4.12):

$$I_C = I_{MOS} + I_{PNP} \quad 4.14$$

$$\Rightarrow I_C = (\beta + 1)I_{MOS} \quad 4.15$$

$$\Rightarrow I_C = (\beta + 1)K_P \left[(V_{GS} - V_T)V_{DS}^* - \frac{V_{DS}^{*2}}{2} \right] \quad 4.16$$

$$\Rightarrow I_C = (\beta + 1) \frac{\mu_C Z C_{ox}}{L} \cdot \left[(V_{GS} - V_T)V_{DS}^* - \frac{V_{DS}^{*2}}{2} \right] \quad 4.17$$

Avec, IMOS : courant drain du MOS (T3), expression rapprochée [64] d'un courant de MOS en phase ohmique,

IPNP : courant collecteur du transistor PNP (T1).

β : gain statique en courant du transistor PNP

VGS : tension grille source du transistor MOS ou tension VGE

VT : tension de seuil du transistor MOS

KP : paramètre de transconductance du canal MOS [A/V²], donné par :

Z et L : respectivement largeur et longueur du canal MOS

μ_C : la mobilité des électrons dans le canal MOS [m²/(V.s)]

Cox = ϵ_{ox}/W_{ox} : capacité de l'oxyde de grille par unité de surface [F/m²]

ϵ_{ox} : permittivité de l'oxyde ($\epsilon_{ox} = 3,41 \cdot 10^{-11}$ F/m pour la silice)

Wox : épaisseur de l'oxyde

V*DS : tension drain source « intrinsèque » au MOS, elle illustre la zone restreinte du canal N

$$V_{DS}^* = V_{DS} - (R_D + R_{MOS}) \cdot I_{MOS} \quad 4.18$$

Avec, RD : résistance de la zone épitaxie N-.

RMOS : résistance intrinsèque du canal MOS.

VDS : tension de drain source « externe » au MOS ou tension VBE,T1 de polarisation du transistor NPN.

Une phase de saturation. VCE > Vp, le courant IC est quasi constant. La diffusion des porteurs de charge est limitée à cause du pincement du canal MOS [64], suite à l'augmentation du potentiel du drain ou de la source du transistor MOS (figure). La tension VCEsat est la tension de pincement (ou de saturation) de l'IGBT. Au-delà de VCEsat, le courant IC devient pratiquement indépendant de VCE et n'est plus fixé que par la tension VGE. L'expression rapprochée du courant collecteur est alors égale à [64] :

$$V_{CEsat} = V_{BE,T1} + I_{MOS} \cdot (R_D + R_{MOS}) \quad 4.19$$

Avec VBE,T1 : la tension base-émetteur de T1;

RD : la résistance de la couche épitaxie N- (drift), modulé en conductivité;

RMOS : résistance intrinsèque du canal de MOS.

Ce régime présente des contraintes thermiques qui peuvent aboutir à la destruction de l'IGBT si la tension VCE est poussée au-delà de sa limite : VCEsat.

La phase de pincement du canal MOS est appelée phase de « désaturation » en tension de l'IGBT par les électroniciens et les ingénieurs. Pour la suite du manuscrit, le terme « désaturation » désignera donc le pincement de l'IGBT.

- **A l'état bloqué**

L'ouverture de l'IGBT est obtenue lorsque la tension VGE est inférieure à la tension de seuil V_T du transistor MOS T3 (figure). Dans ce cas, il n'y a pas de création de couche d'inversion entre l'émetteur et le collecteur. La tension de blocage VCE est supportée par la jonction des semi conducteur N- et P+ de la figure.

Après le fonctionnement de base de l'IGBT, nous pouvons maintenant étudier les différentes contraintes qui fragilisent sa fiabilité.

4.3.2. Partie Commande

4.3.2.1. Carte de Commande dSPACE – DS 1103

La carte dSpace DS 1103, est une carte single board [66], elle est idéale pour les applications pédagogiques ou de recherche à budget limité puisqu'elle est disponible en ACE kit, c'est-à-dire en kit éducation à des prix très attractifs.



Figure 4.13 : Carte dSPACE DS 1130

Elle dispose de deux DSP, un DSP (Master) de calcul en virgule flottante et un DSP (Slave), en virgule fixe, qui s'occupe de la PWM et des entrées sorties numériques. Elle intègre, en ce qui nous concerne, des convertisseurs analogiques-numériques (ADC), numériques-analogiques, interfaces série, des timers etc [66].



Figure 4.14 : Ports de communication
Carte dSPACE DS 1130

La DS1103 est une carte polyvalente pour le prototypage rapide de lois de commande. Vous pouvez la monter dans une dSPACE Expansion Box ou une dSPACE AutoBox pour tester les fonctions de votre contrôleur en laboratoire ou directement en véhicule. Sa puissance de calcul et ses E/S rapides sont primordiales pour les applications impliquant de nombreux actionneurs et capteurs. Utilisée avec la Real-Time Interface (RTI), la carte contrôleur est entièrement programmable à partir de l'environnement de schéma-bloc Simulink®. Vous pouvez configurer toutes les E/S graphiquement en utilisant la RTI. C'est une façon simple et rapide d'implémenter vos fonctions de commande sur la carte. Le nombre sans égal de ses interfaces d'E/S fait de la DS1103 une carte contrôleur polyvalente destinée à de nombreuses applications. Elle fournit un grand choix d'interfaces comprenant 50 bits de canaux d'E/S, 36 canaux A/N et 8 canaux N/A. Pour les tâches d'E/S supplémentaires, une unité contrôleur DSP, construite autour du DSP TM320F240 de Texas Instruments est utilisée comme sous-système [66].

L'architecture interne de cette carte est illustrée ci-dessous figure 4.15 [66].

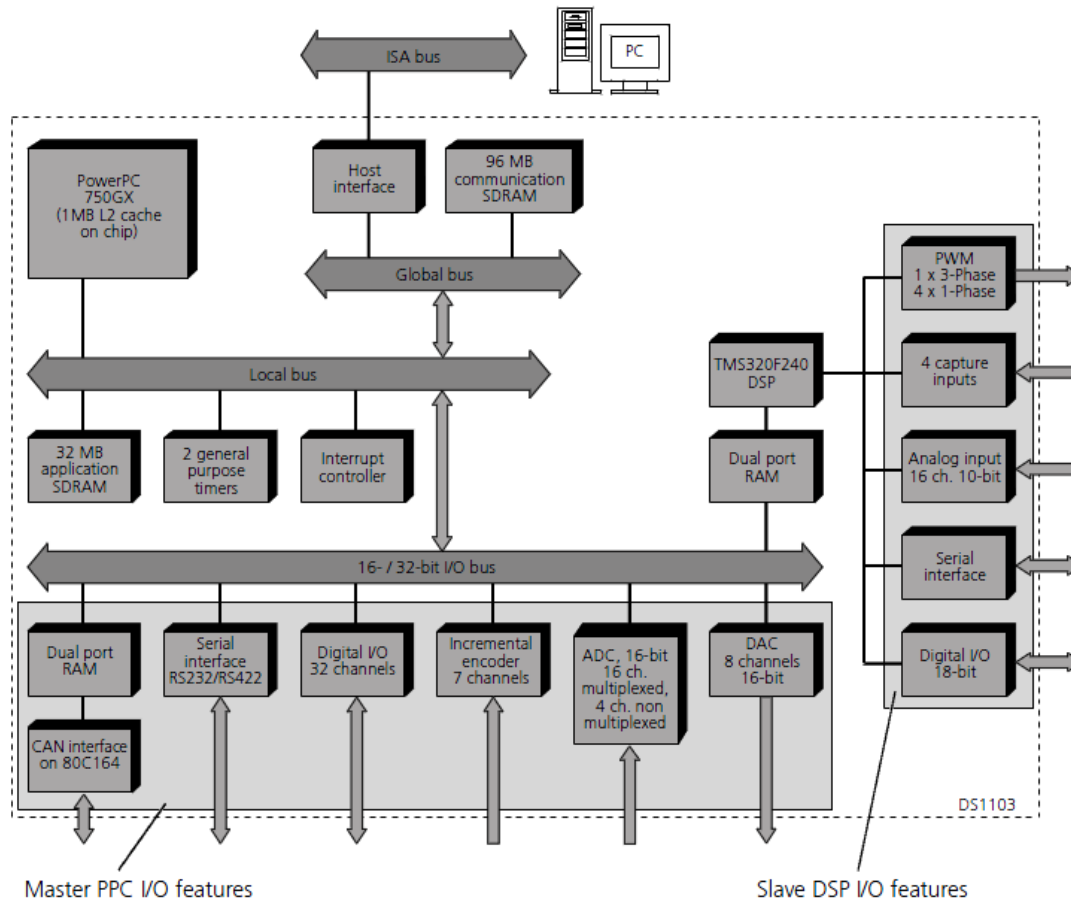


Figure 4.15 : Architecture interne
Carte dSPACE DS 1130

4.3.2.2. Drivers d'IGBT

Le driver est une carte d'interface entre une commande électronique consistant généralement en signaux logiques et un composant de puissance (ou, plus généralement, un tripôle, formé de deux transistors duaux) [64].

Les caractéristiques essentielles d'un driver peuvent être résumées comme suit :

Isolement galvanique entre le signal de commande, venant de la carte du « processeur » (microcontrôleur ou DSP), d'une part, et le composant de puissance d'autre part. Cette isolation permet la protection de la carte du processeur contre l'intervention inopportune de la haute tension ou de courant trop fort du convertisseur. L'intérêt de l'isolement est la possibilité de transmettre des signaux pour des composants fonctionnant à des potentiels différents.

Retard à la transmission du signal de commande le plus faible possible, afin de permettre le fonctionnement en haute fréquence ;

Montage de sortie du driver pour que l'effet Miller du transistor soit complètement surmonté ; il est alors nécessaire que le courant fournie/reçu à/de la grille puisse charger/décharger la capacité équivalente d'entrée le plus rapidement possible ;

Mise en œuvre d'un temps « mort » réglable et compatible avec les durées ton et toff de chaque composant. Nous avons plusieurs types de drivers, et celui que nous avons utilisé est le SHKI22 à transformateur.

Description du driver type SHKI22 à transformateur

C'est une interface de commande pour thyristors et transistors IGBT. Ce driver est composé de deux parties complémentaires pour la commande de 2 éléments.

C'est un composant à technologie hybride, qui contient un générateur d'impulsions, un transformateur qui assure la transmission de la commande et une alimentation isolée des transistors qui attaquent la grille des transistors de puissance.

Un montage à double commande (par deux résistances R1) assure l'attaque de la grille de chaque transistor. Le montage type d'utilisation est présenté à la figure ci-dessous (alimentation +15 V coté carte processeur). Les deux entrées Vin1 et Vin2 sont en principe indépendantes, mais un verrouillage interdit une commande simultanée des deux transistors. Une résistance de « pull up » R est nécessaire. La tension VCE « collecteur-émetteur » (où drain-source) de chaque transistor est contrôlée en permanence ; elle doit être inférieure à 0.7 V lorsque le transistor est en état de fermeture. En cas de court-circuit ou de surintensité dans la charge du convertisseur, un signal « error » est déclenché et une mémoire évite que les deux transistors soient de nouveau mis en conduction. Il faut alors remettre à « 0 » les deux entrées Vin1 et Vin2 pour redémarrer le fonctionnement. Le signal « error », à l'état haut, indique l'état de fonctionnement du driver.

Une tension VCEstat est fixée pour la remise en route du transistor. La « surveillance » de VCE est réglée grâce au circuit RCEC. La valeur de VCEstat à régler est donnée approximativement par la relation :

$$V_{CEstat} (V) = \frac{9 R_{CE} (k\Omega) - 25}{10 + R_{CE} (k\Omega)} \quad 4.20$$

(La valeur typique de VCEstat est 5.6 V)

Les valeurs couramment utilisés sont RCE=24kΩ et C=330 pF.

L'isolement galvanique est assuré par un transformateur d'impulsions fonctionnant à 1 MHz.

La fréquence maximale des signaux de commande est de 100 kHz.

Un réglage du temps mort est prévu grâce aux résistances R_{TD} , selon la relation :

$$t_{TD}(\mu s) = 2,7 + 0,13R_{TD}(k\Omega) \quad 4.21$$

Les résistances R_1 « d'attaque » de la grille doivent être suffisamment faibles pour permettre les pointes de courant et éviter le risque d'oscillation du transistor MOS. En pratique, elles sont choisies supérieures à $3,3 \Omega$ (valeur typique 22Ω). On remarque que les sorties GON1 et GON2 sont réservées pour la commande en fermeture de T1 et de T2, tandis que les sorties GOFF1 ou GOFF2 sont réservées pour la commande en ouverture de T1 et de T2.

La durée de la transmission de la commande en fermeture est de $1 \mu s + t_{TD}$.

La durée de la transmission de la commande à l'ouverture est de $1 \mu s$.

Ce type de driver assure un véritable isolement galvanique, pouvant subir une différence de potentiel entre entrée et sortie qui peut atteindre 2500 V en valeur maximale.

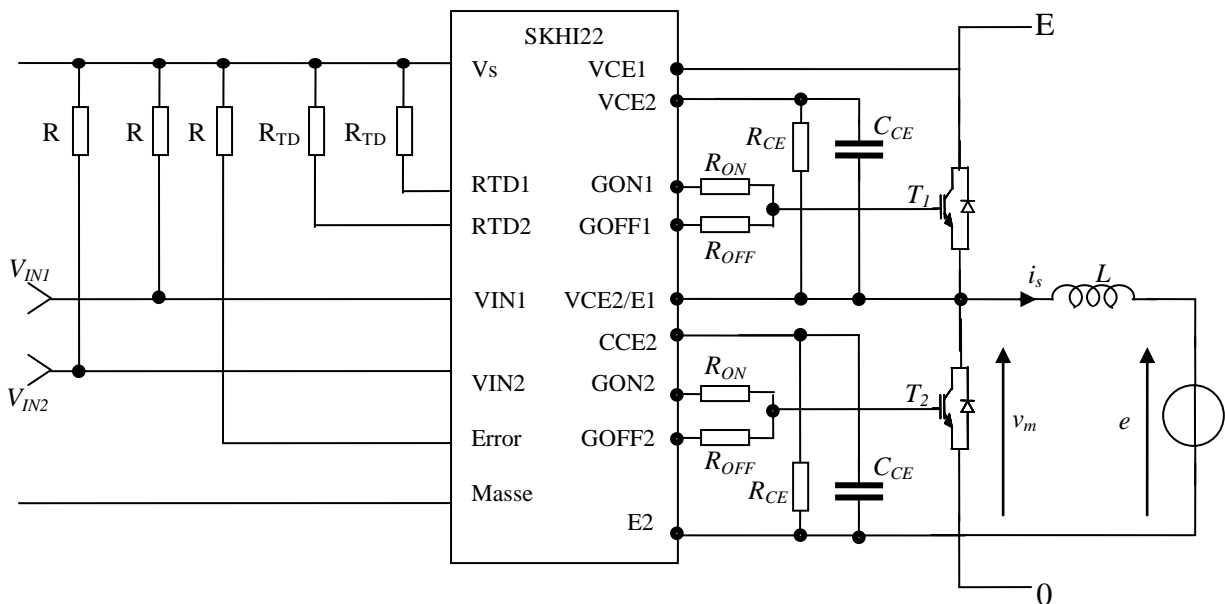


Figure 4.16 : Montage typique du composant hybride SHKI22 pour une commande d'un tripôle formé de transistors duaux.

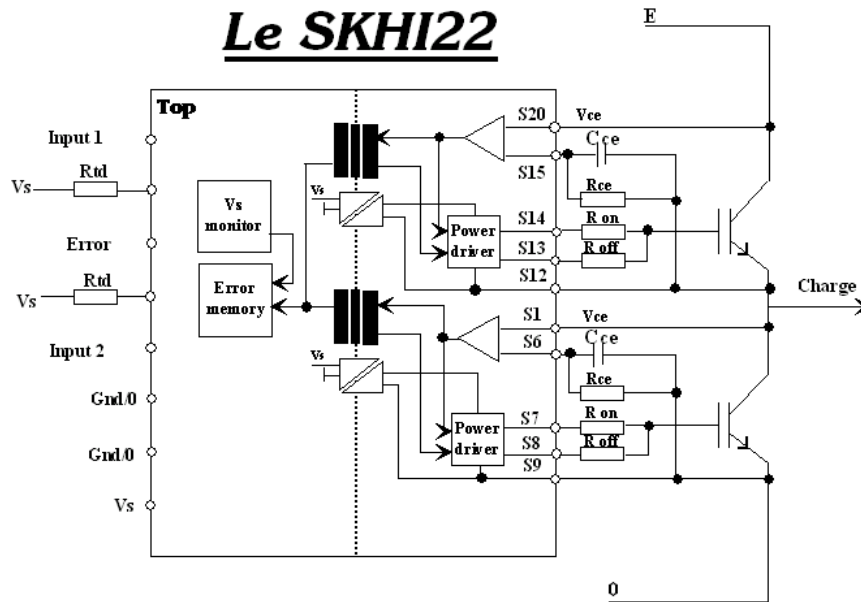


Figure 4.17 : Architecture interne du composant hybride SHKI22.

Il sert à la fois à transmettre aux IGBT les ordres envoyés par la commande, à protéger la commande contre de possibles énergies destructrices provenant des IGBT (isolation galvanique), à protéger les IGBT qu'il pilote par une limitation de leurs courants (détection de surintensité) collecteur, qu'il mesure (figures 4.16 et 4.17) et compare à un seuil limite à ne pas dépasser.



Figure 4.18 : Boîtier SHKI22.

Malgré toutes ses capacités de protection, un driver est généralement limité pour isoler entièrement un bras d'onduleur en cas de problème de phase.

4.3.2.3. Circuit d'adaptation des signaux de commande

La figure 4.19 montre le schéma d'une simple interface utilisée pour adapter les niveaux de tensions des signaux de commandes du microcontrôleur (0 V et 5 V) à ceux requis par les interrupteurs (-15 V et + 15 V). Un comparateur est utilisé pour convertir les niveaux de tensions.

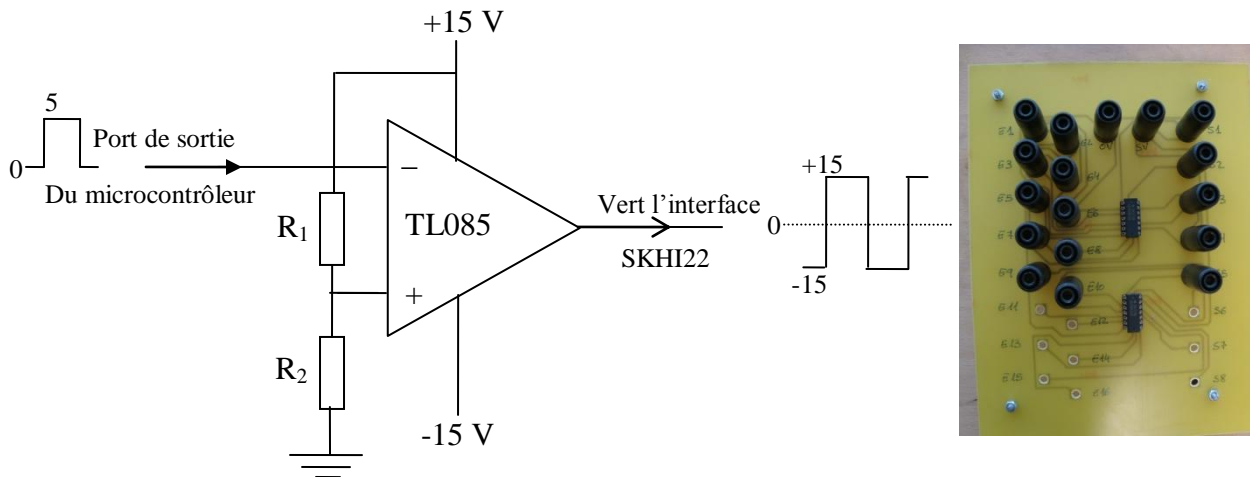


Figure 4.19 : Adaptateur du niveau de la tension de commande.

4.3.2.4. Capteur de vitesse et de Couple de rotation

- Le capteur de vitesse est une dynamo tachymétrique qui fournit une tension proportionnelle à la vitesse de rotation. Celle ci, est montée sur un support de fixation, est équipée d'un 1/2 manchon permettant de l'accoupler rapidement sur le 2ème bout d'arbre de la machine d'essai.

Constante de vitesse : 0,02 V par tour minute, f.e.m à 1 000 min⁻¹ : 20 V .

I maxi. : 0,15 A / Vitesse maxi. : 10 000 min⁻¹ / R'induit : 51 Ω .

La boîte à borne de la DT dispose de 2 sorties :

- l'une sur bornes de sécurité de 4 mm donnant 20V à 1000 min⁻¹ .
- l'autre sur fiche DIN donnant 20V à 1000 min⁻¹ .



Figure 4.20 : Capteur de vitesse

- Capteur de couple

Le capteur de couple est caractérisé par :

$C_n = 50 \text{ N.m}$

$N_{max} = 3000 \text{ tr/min}$

Pour visualiser le couple de la machine en utilise le Module de mesures mécaniques



Figure 4.21. Photo capteur de couple

4.3.3. Partie Software

Afin de nous permettre d'exploiter la partie hard, une manipulation logicielle doit être utilisée. Pour programmer et implémenter nos lois de commande proposées, il faut réaliser son schéma de commande sur la boîte à outils Simulink de Matlab (figure). Une bibliothèque spéciale « DS1103 Real Time Library » regroupe les différentes fonctions utiles pour gérer le DSP (figure 4.22). A l'aide de la commande Build dans Simulation Paramètres, on charge le

code directement dans le DSP. Il est préférable de se placer dans son répertoire de travail pour construire le fichier sur Simulink car la construction génère de nombreux fichiers. Le logiciel Control Desk offre une interface qui permet de visualiser et de modifier en temps réel différentes variables du schéma Simulink [66].

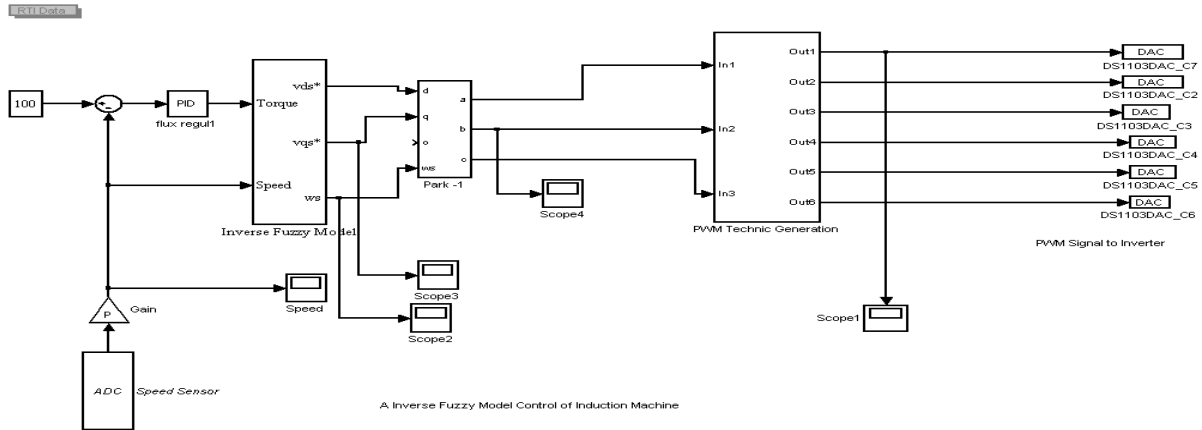


Figure 4.22 :Modèle de commande par modèle inverse flou
Commande en temps réels

ControlDesk est un logiciel d'expérimentation pour le développement direct du contrôleur. Il exécute toutes les tâches nécessaires et vous propose un seul environnement de travail du début à la fin de votre expérimentation. Control Desk Standard peut être exploité d'après deux modes : le mode Développer vous octroie toutes les fonctionnalités et le mode Operator protège vos expérimentations contre des modifications non autorisées [66].

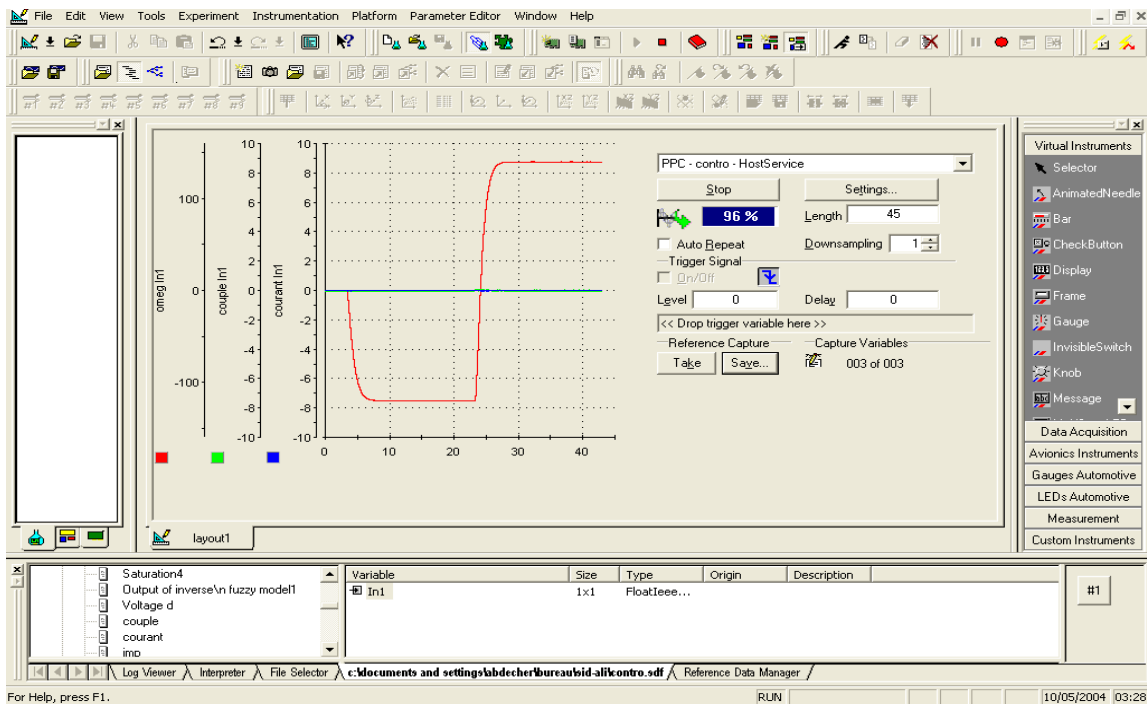


Figure 4.23 : Control Desk

Le contrôle de simulations Simulink est intégré et vous permet de valider hors ligne vos modèles du contrôleur, avec la possibilité sur demande de permuter dans l'univers temps réel dSPACE et vice-versa. Vous pouvez utiliser le même environnement outil pour l'instrumentation virtuelle, l'automatisation et la gestion du jeu de paramètres sans avoir à y apporter de modifications. L'interface utilisateur sophistiquée et les techniques de layout telles que le glisser-dépasser, les menus contextuels, les fenêtres flottantes, les fonctions clavier améliorées et l'aide en ligne complète vous seront très vite familières.

4.4. Structure générale de Banc Expérimental

Le banc d'essais complet à base d'une carte dSpace DS 1103, a été conçu dans le but de pouvoir tester différentes lois de commande du variateur et surtout différentes stratégies de MLI (Figure 4.24). Le premier avantage du banc et la possibilité de compiler les schémas de simulation Matlab/Simulink pour une « simulation temps-réel », c'est à dire de permettre l'essai de la commande du variateur, réalisée sur l'ordinateur, sur un moteur.

Le banc a été aménagé dans le cadre de nos travaux. La carte d'interfaçage entre le panneau de connecteurs du système dSpace et la carte de contrôle du variateur a été réalisée.

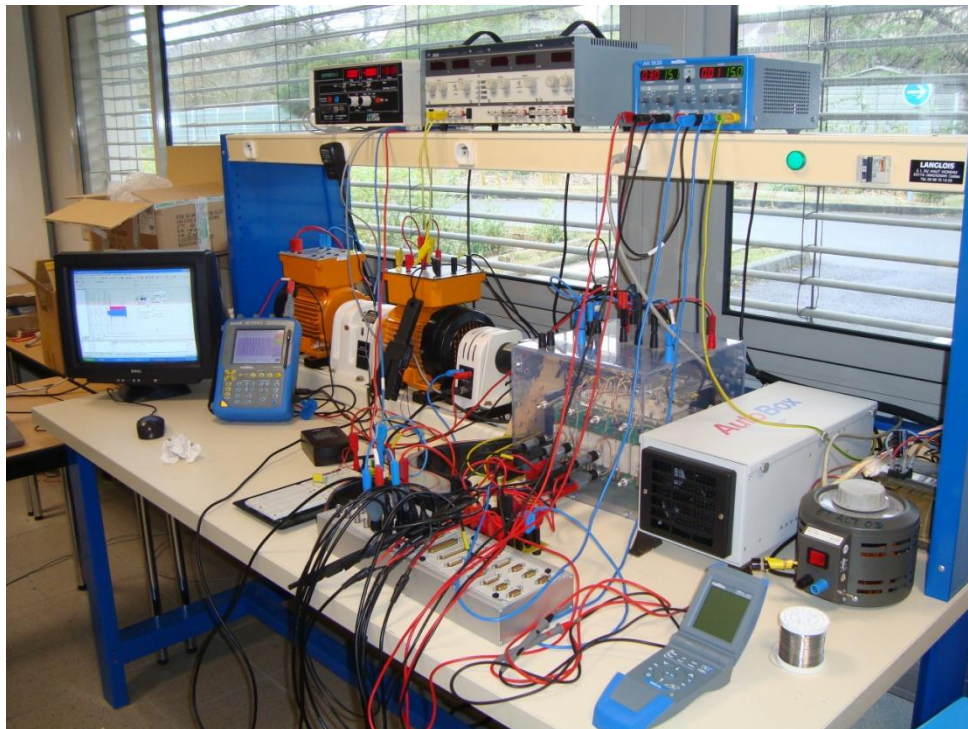


Figure 4.24 : Banc d'implémentation à base d'une carte dSPACE DS 1103

Les éléments apparents du banc (Figure 4.24) :

- Moteur
- Panneau de connecteurs
- Convertisseur (Redresseur+Onduleur)
- AutoBox dSPACE dans laquelle se trouve la carte dSPACE
- Module de mesures mécaniques Réf : MOD'MECA
- Capteurs de courants
- Capteurs de tensions

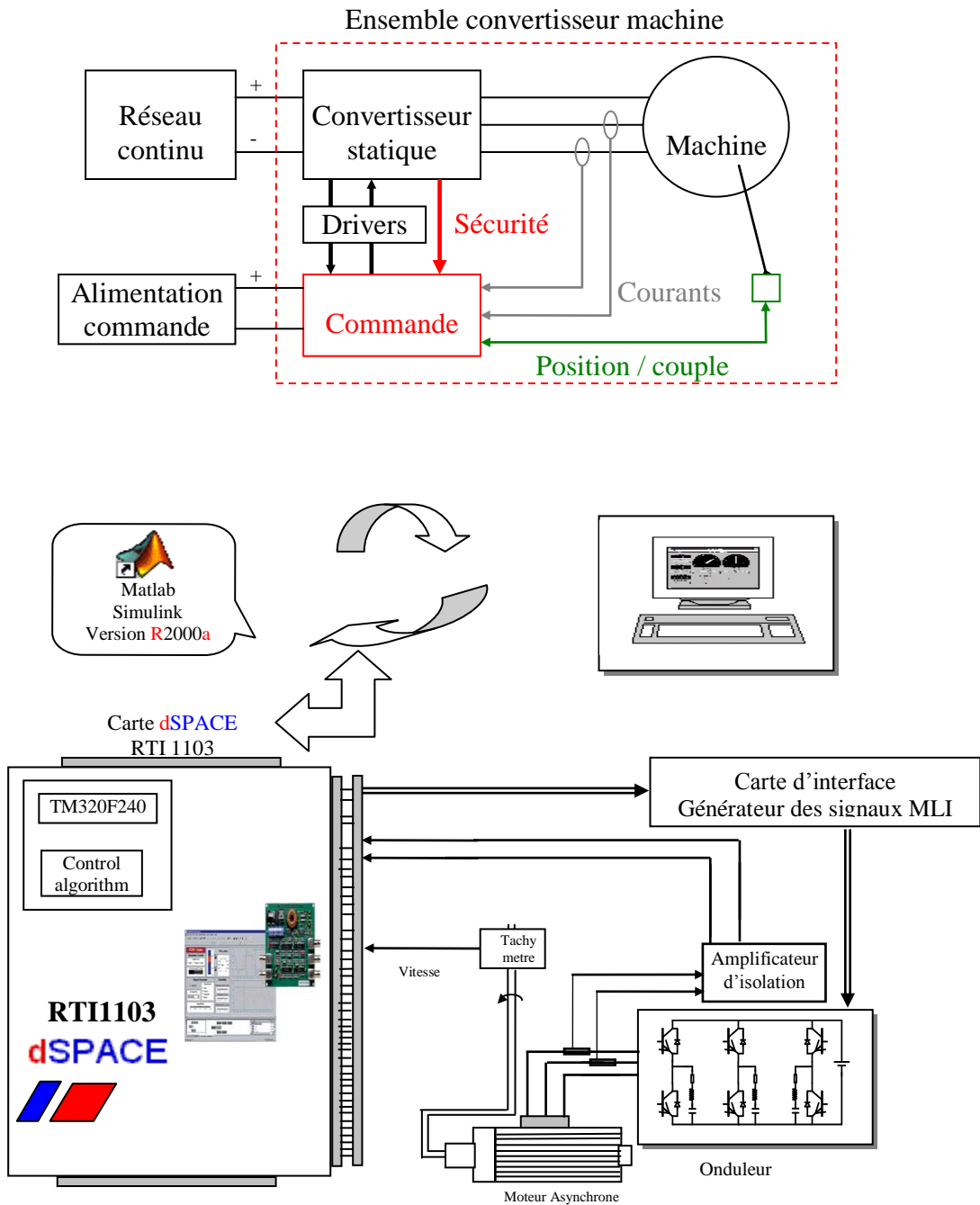


Figure 4.25 : structure général d'une motorisation électrique

Afin d'évaluer les performances des différentes commandes proposées basées sur le modèle inverse flou (MIF) du moteur à induction, des évaluations expérimentales ont été réalisées. Dans notre cas, nous avons tenu compte des problèmes dus à la commande numérique, en particulier, l'effet d'échantillonnage. Quand la période d'échantillonnage reste suffisamment faible par rapport à la plus petite des constantes de temps du système, la vitesse reste bien contrôlée. Lorsque la période d'échantillonnage devient comparable à celle des constantes de temps on perd le contrôle de la vitesse son régime transitoire.

A chaque période d'échantillonnage, la vitesse du rotor est acquise, ensuite les tensions de références V^*_{ds} et V^*_{qs} sont obtenues à l'aide du modèle inverse flou de la machine pour toutes les lois de commande basées sur ce modèle. Les tensions de référence des trois phases obtenues après transformation sont envoyées au générateur des signaux MLI qui fournit les séquences d'alimentation des trois phases en utilisant les nouvelles valeurs de référence. Le fonctionnement se poursuit, mais avec les nouvelles tensions et le circuit de commande attend la fin de la période d'échantillonnage pour recommencer avec de nouvelles valeurs.

Afin de valider expérimentalement les résultats obtenus en simulations précédentes sur la commande de la machine à induction, des évaluations expérimentales ont été réalisées. On a utilisé les conditions d'essai que la simulation.

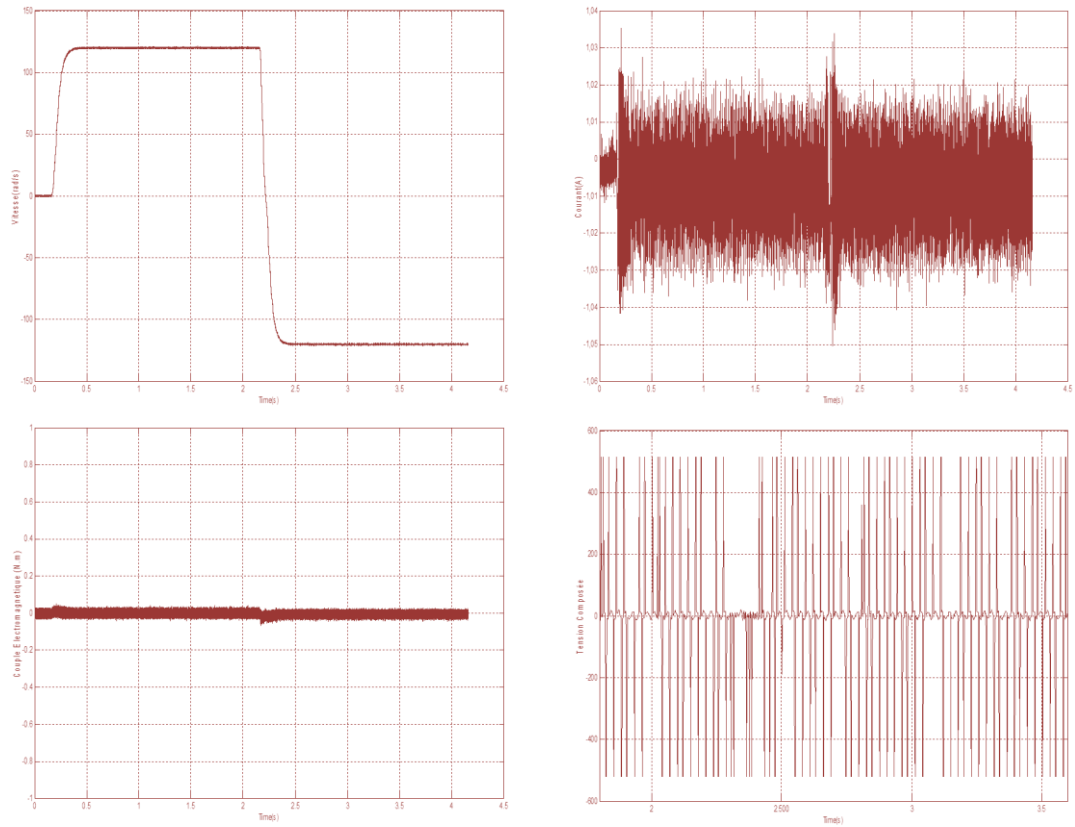


Figure 4.26. Les résultats d'expérimentations de l'approche classique

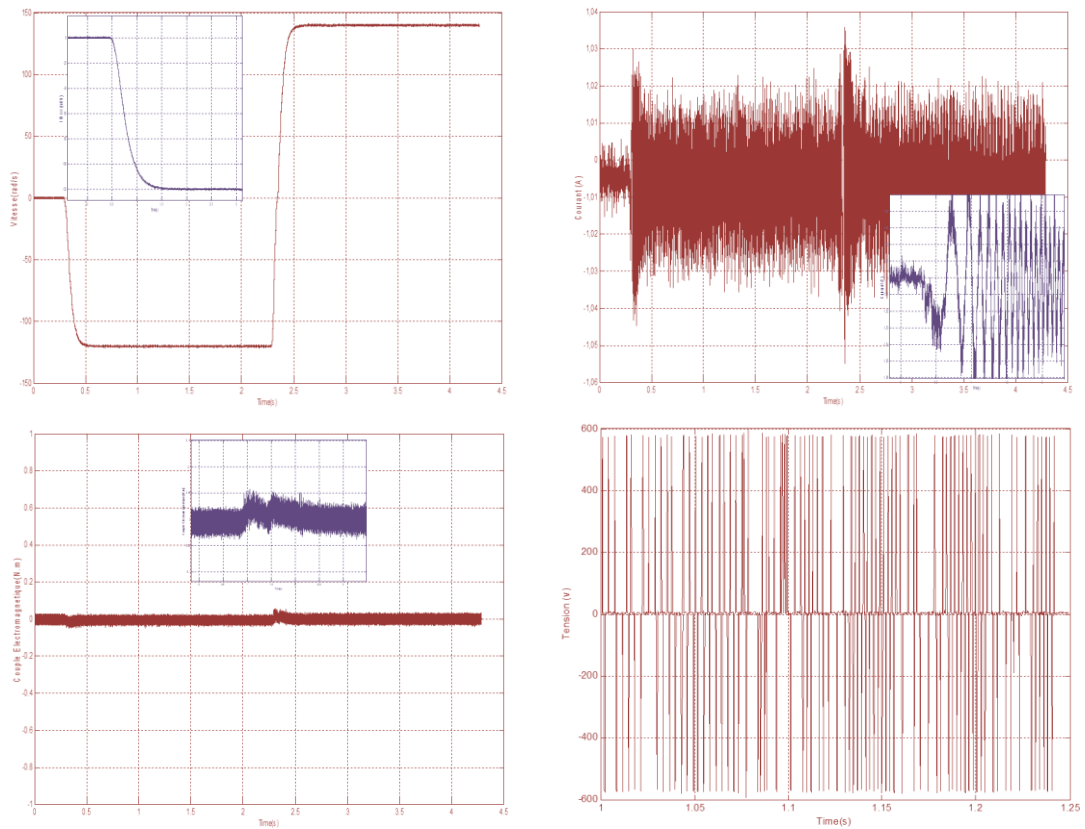


Figure 4.27. Les résultats d'expérimentations de l'approche intelligente

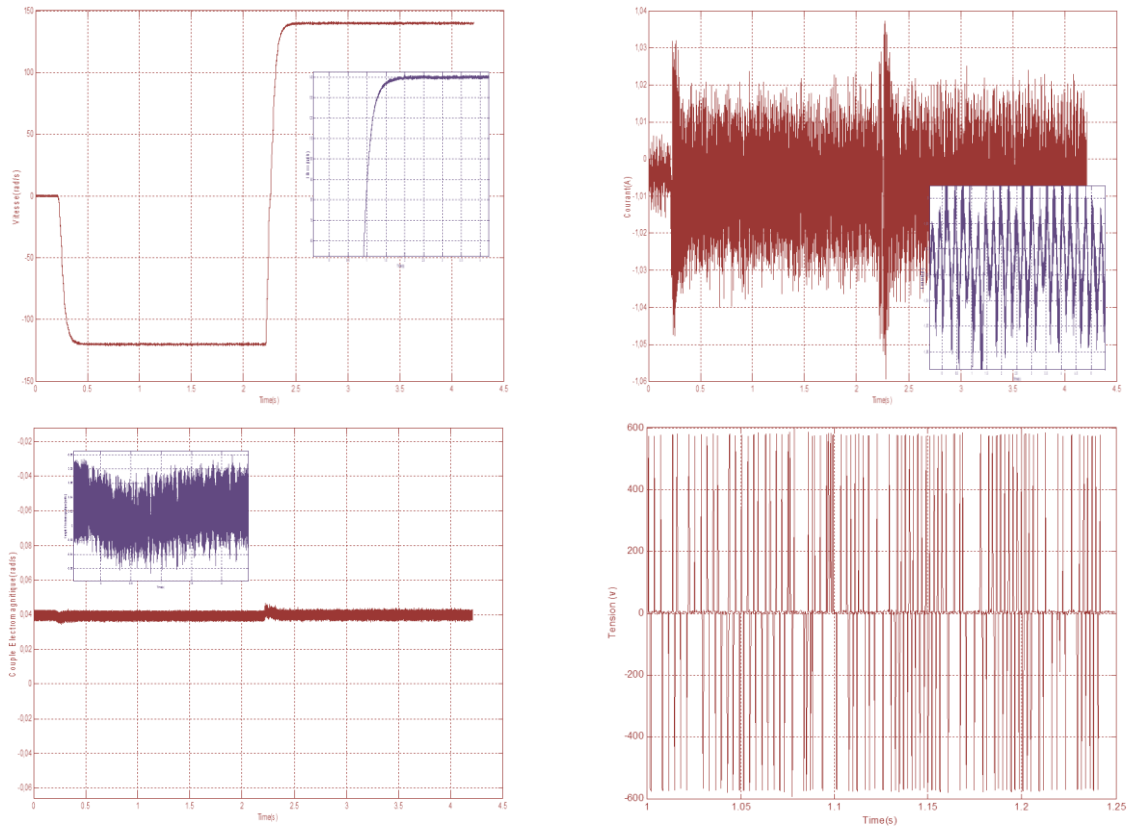


Figure 4.28 Les résultats d'expérimentations avec un régulateur IP anti-saturation

4.5 Conclusion

Ces expérimentations ont permis de comparer et de vérifier le nouveau modèle inverse flou et les lois de commande de la machine asynchrone. Nous avons vérifié tout d'abord le bon fonctionnement de tous les capteurs dans toute la plage de fonctionnement, les trois bras de l'onduleur ainsi que les ports d'entrées et de sorties de la carte dSPACE, puis nous avons adapté les signaux de commande et finalement implémenter les différentes lois de commande.

Conclusion Générale

Dans cette thèse, le travail présenté apporte une nouvelle contribution dans la modélisation intelligente des systèmes incertains appliquée au moteur asynchrone.

Nous avons vu les étapes successives permettant d'élaborer une commande intelligente basée sur un modèle intelligent inverse du moteur asynchrone en utilisant les avantages de la modélisation floue, jusqu'à sa mise en œuvre expérimentalement.

En ce qui concerne la commande par flux orienté celle-ci a montré son efficacité du point de vue découplage, dans le régime permanent. Néanmoins, l'inconvénient majeur déjà rencontré et qui concerne la sensibilité vis-à-vis des incertitudes inhérentes au système, des perturbations externes subite toujours et à l'implémentation pratique de cette loi de commande. A cette effet, notre objectif de travail consistait à voir les choses autrement que d'habitude, c.à.d. d'essayer de trouver un meilleur moyen pour concevoir un modèle intelligent qui remplace le modèle mathématique d'orientation du flux rotorique, c.à.d. le modèle mathématique inverse de la machine asynchrone, et qui permet de surmonter ces problèmes et ce, sans toucher au régulateur classique. A cet effet, et puisque il s'agit d'une modélisation, nous avons étudié le concept de la modélisation intelligente floue d'une manière générale. Par la suite, une approche de modélisation floue introduite par Ben-Ghalia applicable sur les systèmes incertains a été appliquée au moteur asynchrone. Cette approche permet de convertir une représentation analytique du système incertain en une représentation basée sur des règles linguistique de type « if-then ». Elle offre également l'avantage d'être adapté à la conception des systèmes de commande robuste du fait quelle tient compte des plages de variation des paramètres internes de la machine.

Pour bénéficier des avantages que procure cette technique, et vu la complexité, la non-linéarité et les incertitudes du modèle du moteur, le fruit de notre travail de doctorat s'est concrétisé par une contribution à la modélisation inverse floue. A cet effet, nous avons envisagé d'adapter cette technique, afin de concevoir un modèle inverse flou de la machine asynchrone, et qui tient en compte la condition d'orientation du flux rotorique, sachant quelle consiste en une nouvelle approche de commande des processus incertains par leurs modèles inverse flou en général, et de commande de la machine asynchrone triphasée en particulier. Ainsi, cette technique de modélisation floue sert à approximer le système par une représentation floue tout en tenant compte de la description analytique grossière du modèle considéré et des incertitudes qui pourraient survenir. Le modèle ainsi obtenu n'est pas

constitue d'une combinaison entre expressions réelles et floues, mais il est constitué intégralement à base de règles floues. La robustesse pour ce genre de commande est toujours assurée. Les résultats, et de simulation, et de l'expérimentation effectués ont montré que l'approximation du modèle inverse mathématique réel par un modèle inverse flou est assuré.

Afin de tirer au profit de cette nouvelle technique de modélisation, d'autre type de régulateur ont été appliqué au système basé sur le modèle inverse flou.

Nous attirons l'attention sur le fait que le réglage obtenu du système basé sur le modèle inverse flou pour le régulateur IP est assuré, L'anti-saturation ne rentre en jeu que lorsque le régulateur se retrouve en butée de courant. Par contre, il est plus rapide et sollicite beaucoup plus l'actionneur pour rejeter plus vite la perturbation qu'un régulateur classique PID.

Pour bénéficier des avantages de la logique floue, et vu la complexité, la non-linéarité et les incertitudes du modèle du moteur, une technique de commande à structure variables floue a été introduite au modèle inverse flou, afin de concevoir un régulateur qui résiste à toutes variations paramétriques.

La logique floue type-2 a préoccupé beaucoup de chercheurs ces dernières années, et elle a été appliquée dans plusieurs domaines tels que la modélisation vidéo, classification, diagnostic, intelligence artificielle, canaux de transmission, filtres adaptatifs, robots mobiles, etc....

Par nos contributions, nous avons voulu appliquer cette nouvelle logique floue type-2 à la commande adaptative de la machine asynchrone basée sur un modèle inverse flou.

Cette technique a été utilisée au développement d'un contrôleur adaptatif à base de logique floue type-2, où les conditions de sa mise en œuvre ont été suffisamment étudiées. La stabilité de la boucle de commande a été analysée par la méthode de Lyapunov. Comme il a été constaté dans les discussions, les résultats de simulation obtenus par cette méthode ont prouvé l'efficacité de cette approche.

Au niveau de l'implantation de la commande, nous avons implémenté nos lois de commande utilisé une carte dSPACE DS 1130, qui nous a permet de nous donner des résultats très satisfaisants.

Suggestions et perspectives

La méthode proposée est applicable sur une large classe de systèmes incertains. Son choix repose sur la présence analytique, qui peut être non-linéaire et de représentation complexe. Par contre, le nombre d'entrées du modèle de commande, 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. Une fois le modèle flou réalisé, un réglage postérieur peut être effectué par des algorithmes d'optimisation dédiés pour les systèmes flous, tels que la programmation évolutionnaire, en particulier les algorithmes génétiques, et les algorithmes adoptés dans les systèmes neuronaux qui nécessitent de convertir le système flou en une architecture neuro-floue.

Une modélisation floue type-2 pourrait être appliquée, afin d'améliorer la précision du modèle intelligent, et une amélioration de l'alimentation électrique du système à travers l'introduction des convertisseurs multi niveaux à hautes performance pourrait être appliqué.

Annexes

Annexe A

A.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 A.1.

– Les fonctions triangulaire et trapézoïdale qui sont les plus fréquemment utilisées :

$$\text{trimf}_A(x; [a \ b \ c]) = \max\left(\min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0\right) \quad \text{A.1}$$

$$\text{trapmf}_A(x; [a \ b \ c \ d]) = \max\left(\min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right) \quad \text{A.2}$$

– La fonction gaussienne de probabilité utilisée dans les problèmes nécessitant une adaptation de ses paramètres :

$$\text{gaussmf}_A(x; [\sigma \ x_0]) = \exp\left(-\frac{(x-x_0)^2}{\sigma^2}\right) \quad \text{A.3}$$

– La fonction en forme de cloche, définie par :

$$\text{bellmf}_A(x; [a \ x_0]) = \frac{1}{1 + \left(\frac{x-x_0}{a}\right)^2} \quad \text{A.4}$$

– La fonction en forme de cloche généralisée, qui est une extension de la fonction de distribution de probabilité de Cauchy [143], définie par :

$$\text{gbellmf}_A(x; [a \ b \ x_0]) = \frac{1}{1 + \left(\frac{x-x_0}{a}\right)^{2b}} \quad \text{A.5}$$

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 $(x_0 \pm a, 1/2)$.

– La fonction trigonométrique, définie par :

$$\text{trig}_A(x; [a \ x_0]) = \frac{1}{2} \left(1 + \cos \left(\frac{\pi(x - x_0)}{2a} \right) \right) ; \quad x_0 - 2a \leq x \leq x_0 + 2a \quad \text{A.6}$$

– La fonction sigmoïdale :

$$\text{sigmf}_A(x; [a \ x_0]) = \frac{1}{1 + \exp(a(x - x_0))} \quad \text{A.7}$$

La fonction sigmoïdale est souvent utilisée comme fonction d'activation dans les systèmes neuronaux artificiels [143].

– Les fonctions en forme de S :

$$\text{smf}_A(x; [x_0 \ x_1]) = \begin{cases} 2 \left(\frac{x - x_0}{x_1 - x_0} \right)^2 & x_0 < x < \frac{x_0 + x_1}{2} \\ 1 - 2 \left(\frac{x_1 - x}{x_1 - x_0} \right)^2 & \frac{x_0 + x_1}{2} < x < x_1 \\ 0 & x \leq x_0 \\ 1 & x \geq x_1 \end{cases} \quad \text{A.8}$$

où x_0 et x_1 déterminent les points d'arrêt de la fonction d'appartenance.

– Les fonctions en forme de Z :

$$\text{zmf}_A(x; [x_0 \ x_1]) = \begin{cases} 1 - 2 \left(\frac{x - x_0}{x_1 - x_0} \right)^2 & x_0 < x < \frac{x_0 + x_1}{2} \\ 2 \left(\frac{x_1 - x}{x_1 - x_0} \right)^2 & \frac{x_0 + x_1}{2} < x < x_1 \\ 1 & x \leq x_0 \\ 0 & x \geq x_1 \end{cases} \quad \text{A.9}$$

– La fonction π caractérisée par le produit ou le minimum entre les fonctions en forme de s et en forme de z.

$$\pi\text{mf}_A(x; [x_{10} \ x_{11} \ x_{20} \ x_{21}]) = \text{smf}_A(x; [x_{10} \ x_{11}]) \cdot \text{zmf}_A(x; [x_{20} \ x_{21}]) \quad \text{A.10}$$

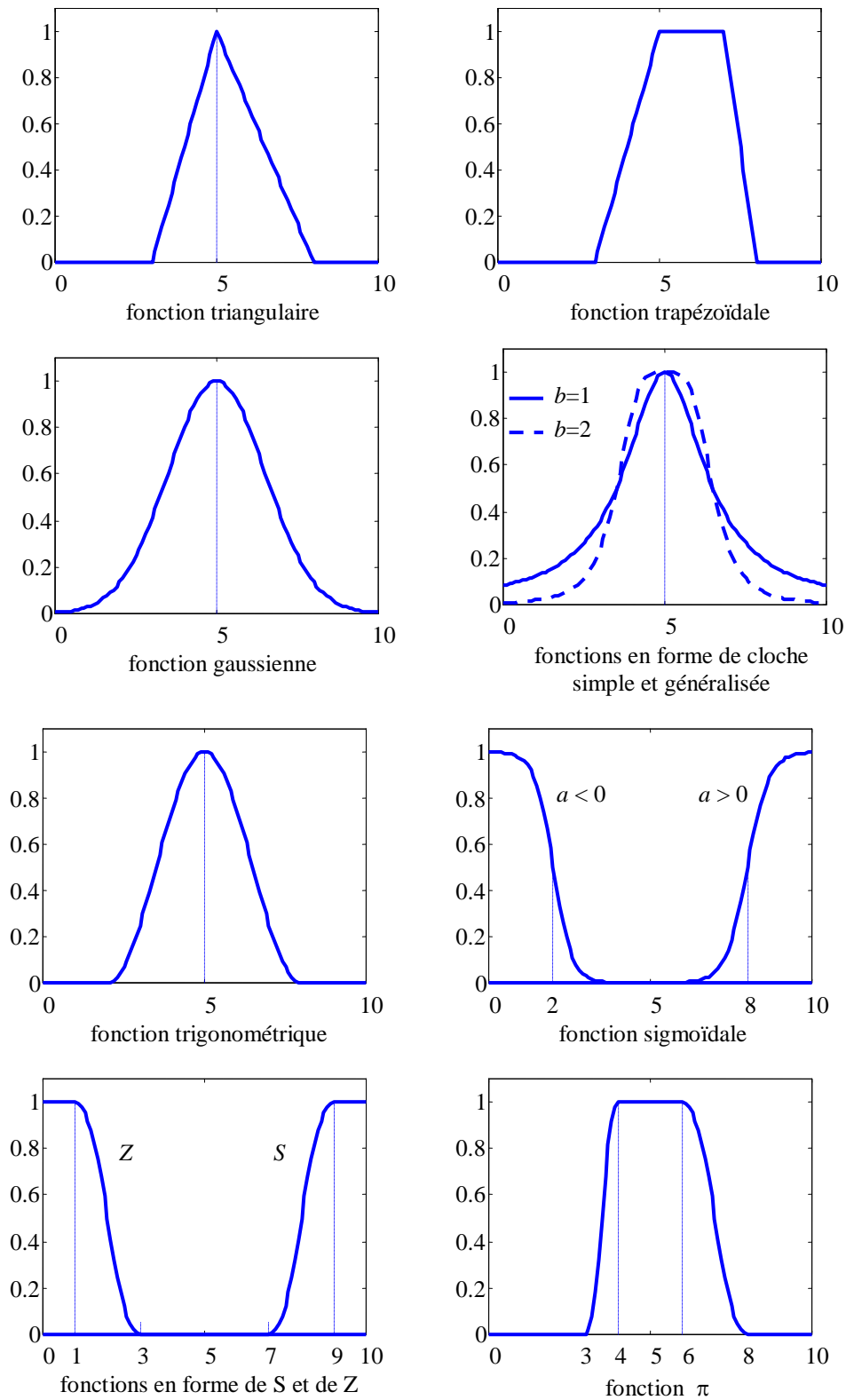


Figure A.1 Formes usuelles des fonctions d'appartenance

A.2. Opérateurs usuels de conjonction, disjonction et de complément

Soient 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 :

$$\begin{aligned}\mu_{A \cap B}(x) &= T(\mu_A(x), \mu_B(x)) \\ \mu_{A \cup B}(x) &= S(\mu_A(x), \mu_B(x))\end{aligned}\tag{A.11}$$

avec $T(\cdot, \cdot)$ satisfait les conditions d'une T-norme [143] définies par :

$$\begin{aligned}T(0,0) &= 0 \\ T(a,1) &= T(1,a) = a \\ T(a,b) &\leq T(c,d) \text{ si } a \leq c \text{ et } b \leq d \text{ (monotonie)} \\ T(a,b) &= T(b,a) \text{ (commutativité)} \\ T(a,T(b,c)) &= T(T(a,b),c) \text{ (associativité)}\end{aligned}\tag{A.12}$$

et $S(\cdot, \cdot)$ satisfait les conditions d'une T-conorme définies par :

$$\begin{aligned}S(1,1) &= 1 \\ S(a,0) &= S(0,a) = a \\ S(a,b) &\leq S(c,d) \text{ si } a \leq c \text{ et } b \leq d \text{ (monotonie)} \\ S(a,b) &= S(b,a) \text{ (commutativité)} \\ S(a,S(b,c)) &= S(S(a,b),c) \text{ (associativité)}\end{aligned}\tag{A.13}$$

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

$$\begin{aligned}N(0) &= 1 \text{ et } N(1) = 0 \\ N(a) &\leq N(b) \text{ si } a \geq b \text{ (monotonie)}\end{aligned}\tag{A.14}$$

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:

loi de De Morgan généralisée	Loi de De Morgan	
$S(a,b) = N[T(N(a), N(b))]$	$A \cup B = \overline{\overline{A} \cap \overline{B}}$	A.15
$T(a,b) = N[S(N(a), N(b))]$	$A \cap B = \overline{\overline{A} \cup \overline{B}}$	

A.3. Quelques exemples des opérateurs de T-norme, de T-conorme et de complément

Le tableau A.1 montre quelques opérateurs de T-norme, de son conjoint T-conorme et du complément, utilisés dans la littérature [143].

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	$a \cdot b$	$a + b - a b$	$1 - a$
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 A.1 : Exemple d'opérateurs de conjonction, disjonction et de complément

ANNEXE C

C.1 DEMONSTRATION DU LEMME (3.1)

Soit \tilde{N} est un ensemble flou dont le support est strictement négatif, c'est à dire :

$$\forall z \in \mathbb{R}^+ : \mu_{\tilde{N}}(z) = 0 \quad \text{C.1}$$

\tilde{e} le vecteur d'erreur flou, $\tilde{s}(\tilde{e})$ le vecteur flou de commutation défini par :

$$\tilde{s}(\tilde{e}) = \begin{pmatrix} \tilde{s}_1(\tilde{e}) \\ \cdots \\ \tilde{s}_m(\tilde{e}) \end{pmatrix} \quad \text{C.2}$$

C.1.1 Lemme : La condition d'atteignabilité de la dynamique correspondant à la $i^{\text{ème}}$ entrée, définie par :

$$\exists \tilde{N} \text{ tel que } \tilde{s}_i(\tilde{e}) \cdot \tilde{\dot{s}}_i(\tilde{e}) \subset \tilde{N} ; i = 1, \dots, m \quad \text{C.3}$$

est équivalente à la condition (C.4) définie par :

$$\begin{aligned} \forall s_i(e) \in \text{Supp}(\tilde{s}_i(\tilde{e})) - \{0\} \text{ et } \forall \dot{s}_i(e) \in \text{Supp}(\tilde{\dot{s}}_i(\tilde{e})) - \{0\} \\ \text{Alors : } s_i(e) \cdot \dot{s}_i(e) < 0 \end{aligned} \quad \text{C.4}$$

Preuve : [65]

Soit un ensemble flou F_{ss} défini par :

$$F_{ss} = \tilde{s}_i(\tilde{e}) \tilde{\cdot} \tilde{\dot{s}}_i(\tilde{e}) \quad \text{C.5}$$

avec : $\tilde{\cdot}$ l'extension de l'opérateur de multiplication sur les ensembles flous.

Le support de F_{ss} sera donc décrit par l'expression :

$$\text{supp}(F_{ss}) = \left\{ s_i(e) \cdot \dot{s}_i(e) \in \mathbb{R} \ / \ s_i(e) \in \text{supp}(\tilde{s}_i(\tilde{e})) \text{ et } \dot{s}_i(e) \in \text{supp}(\tilde{\dot{s}}_i(\tilde{e})) \right\} \quad \text{C.6}$$

Utilisant l'équation (C.5), la condition (C.4) peut être écrite par :

$$F_{ss} \subset \tilde{N} \quad \text{C.7}$$

Utilisant la définition de l'inclusion des ensembles flous [41 et 42], la condition C.7 devient :

$$\forall z \in \mathbb{R} : \mu_{F_{ss}}(z) < \mu_{\tilde{N}}(z) \quad \text{C.8}$$

Si on prend $z_i = s_i(e) \cdot \dot{s}_i(e) \in \text{supp}(F_{ss})$ et $z \geq 0$, les équations (C.8) et (C.3) conduit à :

$$\mu_{F_{ss}}(z_i = s_i(e) \cdot \dot{s}_i(e)) < 0 \quad \text{C.9}$$

Ce qui est une contradiction, car les degrés d'appartenance appartiennent à l'intervalle $[0,1]$.

Donc, nous pouvons conclure que :

$$\forall z_i = s_i(e) \cdot \dot{s}_i(e) \in \text{supp}(F_{ss}) \quad : \quad s_i(e) \cdot \dot{s}_i(e) < 0 \quad \text{C.10}$$

D'autre part, pour que la condition C.10 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.8), (C.9) et (C.10), nous pouvons conclure que pour tout $s_i(e) \in \text{supp}(\tilde{f}_i(\tilde{e})) - \{0\}$ 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

D.1 LES FONCTIONS DE CARATHEODORY

Avant de présenter les fonctions de Carathéodory, quelques définitions de base sont données au préalable [143].

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 \mathfrak{G} de parties de E satisfaisant aux propriétés suivantes :
 - a) la réunion de toute famille d'éléments de \mathfrak{G} appartient à \mathfrak{G}
 - b) l'intersection de toute famille finie d'éléments de \mathfrak{G} appartient à \mathfrak{G}

En particulier, la partie vide et la partie pleine de E appartiennent à \mathfrak{G} .

- On appelle *partie ouverte* (ou un *ouvert*) de l'espace topologique E toute partie de E appartenant à \mathfrak{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 .
- 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:

$$d(x, y) = 0 \Leftrightarrow x = y \tag{D.1}$$

$$d(x, y) = d(y, x) \tag{D.2}$$

$$d(x, z) \leq d(x, y) + d(y, z) \quad \text{D.3}$$

La dernière condition (D.3) 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 :

$$\begin{aligned} d(x, y) &= |x_0 - y_0| + |x_1 - y_1| \\ d(x, y) &= (x_0 - y_0)^2 + (x_1 - y_1)^2 \\ d(x, y) &= [(x_0 - y_0)^2 + (x_1 - y_1)^2]^{1/2} \end{aligned} \quad \text{D.4}$$

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 $P(E)$.
- *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 :

$$\forall A, B \in P(E) \text{ alors } A, B \text{ disjoints} \Rightarrow m(A \cup B) = m(A) + m(B) \quad \text{D.5}$$

- 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 .

D.1.4.1 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 [143].

On appelle mesure de Lebesgue sur \mathbb{R}^n la mesure de Radon :

$$m_x = m_{x_1} \otimes m_{x_2} \otimes \dots \otimes m_{x_n} \quad \text{D.6}$$

où, pour tout élément i de $[1, n]$, m_{x_i} désigne la mesure de Lebesgue sur \mathbb{R} .

D.1.4.2 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 $\mu : \text{KC}(E) \rightarrow \mathbb{C}$ satisfaisant à la condition suivante :

Pour toute fonction positive f appartenant à $\text{KC}(\mathbb{R})$, D.7
 il existe un nombre réel positif M tel que,
 $\forall g \text{ de } \text{KC}(E) : |g| < f \Rightarrow |\mu(g)| \leq M.$

Toute forme linéaire positive sur $\text{KC}(E)$ est une mesure de Radon, dite positive.

D.1.4.3 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 :

$$\|f(x, t)\| \leq C_\Sigma \quad \text{D.8}$$

Références bibliographique

- [1]. Lemos, A. Caminhas, W. Gomide, F. "Multivariable Gaussian Evolving Fuzzy Modeling System", *Fuzzy Systems, IEEE Transactions on*, 91 – 104, Feb. 2011
- [2]. Wu, D. Mendel, J. M. "Linguistic Summarization Using IF–THEN Rules and Interval Type-2 Fuzzy Sets", *Fuzzy Systems, IEEE Transactions on*, 136 – 151, Feb. 2011
- [3]. Stepnicka, M. De Baets, B. Noskova, L. "Arithmetic Fuzzy Models", *Fuzzy Systems, IEEE Transactions on*, 1058 – 1069, Dec. 2010
- [4]. L. Barazane, "Application des commandes émergents au Moteur asynchrone", Thèse de Doctorat d'Etat, ENP Alger, 2003.
- [5]. Prado, R.P. Garcia-Galán, S. Exposito, J. Yuste, A.J. "Knowledge Acquisition in Fuzzy-Rule-Based Systems With Particle-Swarm Optimization", *Fuzzy Systems, IEEE Transactions on*, 1083 – 1097, Dec. 2010
- [6]. Narimani, M. Lam, H.K. "SOS-Based Stability Analysis of Polynomial Fuzzy-Model-Based Control Systems Via Polynomial Membership Functions", *Fuzzy Systems, IEEE Transactions on*, 862 – 871, Oct. 2010
- [7]. P.Bortolet, S. Boverie, A. Titli, "Modélisation floue et contrôle de systèmes non linéaires", LFA'96-Nancy (France), pp.106-133, dec. 1996.
- [8]. M.Ben-Ghali, "Modelling and robust control of uncertain dynamical systems using fuzzy set theory", *Int. J. Control*, vol.68, n°6, pp. 1367-1395, 1997.
- [9]. L. Baghli, "Contribution à la commande de la machine asynchrone, utilisation de la logique floue, des réseaux de neurones et des algorithmes génétiques", thèse de Doctorat en génie électrique, Université Henri Poincaré, Nancy, France, 1999.
- [10]. L. A. Zadeh, *Fuzzy sets, Information and control*, Vol. 8, pp. 338-353, 1965.
- [11]. L. A. Zadeh, *Outline of a new approach to the analysis of complex systems and decision process*, *IEEE Trans. Systems, Man & Cybernetics*, Vol. SMC-1, pp. 28-44, 1973.
- [12]. E. Czogala and W. Perdrycz, *On identification in fuzzy systems and its application in control problems*, *Fuzzy sets and systems*, Vol. 6, pp 73-83, 1981.
- [13]. L.A. Zadeh, "Fuzzy Sets", *Information and Control*, Vol.8, pp.338-353, 1965.
- [14]. M.Ben-Ghali, A.T.Alouani, "Fuzzy modelling of dynamic systems". *Proc. of the 26th Southern Symposium on System Theory*, pp.168-171, 1994.
- [15]. M.Ben-Ghali, A.T.Alouani, "Knowledge-base robust control of complex systems", *Intel. Systems for the 21st Century, IEEE Inter. Conf. on Syst. Man & Cyb.*, Vol.2, pp.977-982, 1995.

- [16]. 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.
- [17]. P.Baranyi, I.M. Bavelaar, R. Babuska, et all. "A method to invert a linguistic fuzzy model", Int. J. of Systems Science, vol. 29, n° 7, pp.711-721, 1998.
- [18]. M.Ben-Ghali, A.T.Alouani, "A fuzzy variable structure approach to feedback regulation of uncertain dynamical systems, with application to robotics", Information Sciences, Vol85, pp.241-273, Elsevier Science, 1995.
- [19]. M.Ben-Ghali, "Nonlinear modeling of uncertain dynamical systems using fuzzy set concepts", In Proc. of the 35th IEEE Conf. on Decision and Control, Kobe, Japan, December 11-13, pp.418-423, 1996.
- [20]. 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, pp.807-812, Kobe, Japan, Dec.1996.
- [21]. 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.
- [22]. Z. Zalila, J. Cucumelle « Logique floue et systèmes d'inférence floue : introduction et propriétés », intellitech, intelligent technologie. 2008.
- [23]. T.Takagi and M.Sugeno, "Fuzzy identification of systems and its applications to modeling and control", IEEE Trans. Syst., Man, and Cybernetics, V.15, pp. 116-132, 1985.
- [24]. Kiriakidis Kiriakos, Fuzzy model-based control of complex plants. IEEE Trans. Fuzzy Syst., Vol.6, n°4, nov. 1998.
- [25]. K. Chafaa, "Structures d'identification et de commande des systemes non lineaires bassées sur les techniques floues ", Thèse de doctorat, Batna, 2006.
- [26]. DS1103 PPC Controller Board, Feature, Realise 5.1, May 2006, dSPACE.
- [27]. M. Ben-Ghali, Modelling and robust control of uncertain dynamical systems using fuzzy set theory, Int. J. Control, vol.68, n°6, pp. 1367-1395, 1997.
- [28]. Xi, Z.; Feng, G.; Hesketh, T." Piecewise Integral Sliding-Mode Control for T-S Fuzzy Systems", Fuzzy Systems, IEEE Transactions on, 65 - 74 , Feb. 2011
- [29]. P.Baranyi, I.M. Bavelaar, R. Babuska, et all. A method to invert a linguistic fuzzy model, Int. J. of Systems Science, vol. 29, n° 7, pp.711-721, 1998.
- [30]. R.Babuska, Fuzzy modeling: Principles, methods and applications, In Advances in Methodology, World Scientific, Singapore, pp.187-220, 1998.

- [31]. M.Setnes, R.Babuska, et al, Similarity measures in fuzzy rule base simplification. IEEE Trans. Syst. Man. & Cyb., Part-B, Vol.28-3, pp.376-386, June 1998.
- [32]. M.Setnes, R.Babuska, H.B.Verbruggen, Complexity reduction in fuzzy modeling. Mathematics & Computers in Simulation, Vol.46, pp.507-516, 1998.
- [33]. J. Abonyi, R. Babuska and F. Szeifert, "Modified Gath-Geva Fuzzy Clustering for Identification of Takagi-Sugeno Fuzzy Models," IEEE Trans. Syst., Man, Cybern. Part B: vol.32, no 5, pp. 612-621, Oct. 2002.
- [34]. P. Y. Glorennec, "Algorithmes d'apprentissage pour systèmes d'inférence floue," Hermes, 1999.
- [35]. S. Nascimento, B. Mirkin and F. M. Pives, "Modeling Proportional Membership in Fuzzy Clustering," IEEE Trans. Fuzzy Syst., vol. 11, no. 2, pp. 173-186, Apr. 2003.
- [36]. 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.
- [37]. M.Setnes, Fuzzy rule-base simplification using similarity measures. M.Sc.Thesis, Delft University of Technology, Delft, Netherlands, July 1995.
- [38]. T.Johansen, Fuzzy model based control: Stability, robustness and performance issues. IEEE Trans. on Fuzzy Systems, Vol.2, n°3, pp.221-234, August 1994.
- [39]. Fayez G. Areeda, Amira Y. Haikala and Reham H. Mohammed , "Adaptive neuro-fuzzy control of an induction motor", Ain Shams Engineering Journal, Elsevier, 25 November 2010.
- [40]. Sugeno M., Yasukawa T., "A fuzzy logic based approach to qualitative modeling", IEEE Trans. Fuzzy Sust., vol.1, pp.7-31, Feb.1993.
- [41]. Shinn-Horng Chen Wen-Hsien Ho Jyh-Horng Chou,"Robust Controllability of T-S Fuzzy-Model-Based Control Systems With Parametric Uncertainties", Fuzzy Systems, IEEE Transactions on, 1324 – 1335, Dec 2009
- [42]. k.Benmansour, A.Tlemçani, M.S.Boucherit, S.Barkati, O.Bouchhida, ' Identification et implémentation de contrôleur flou a gain adaptif d'une machine synchrone a aimants permanents', International Conférence EPE'2002 Bultinul institului Politehnic IASI, Tomul XL VIII(LII) FASC.5, Electrotehnica, Energetica, Electronica, Romania, 2002.
- [43]. . AMAMRA, L. Barazane and M.S. Boucherit, A novel inverse fuzzy model of the field oriented control for induction motor, ELECTROMOTION Journal, vol. 15, No. 3, pp. 131-140, September 2008.
- [44]. S. Amamra, L. Barazane, M.S. Boucherit, "A Novel Inverse Fuzzy Model of the Field Oriented Control for Induction Motor", The 2nd International Conference on Electrical Engineering Design and Technologies (ICEEDT'08), Tunis, Tunisia, Nov 2008».

- [45]. S. Amamra, L. Barazane, M.S. Boucherit, "An Approach for Vector Control of Induction Motor by an Intelligent Model Using Fuzzy Logic", The 3rd International Conference on Electrical Engineering (ICEE'09), Algiers, Algeria, May 2009».
- [46]. S. Amamra, L. Barazane, M.S. Boucherit, "Simulink Model of Inverse fuzzy model Control of Induction Motor", The 6th International Conference on Electrical and Electronics Engineering (ELECO'09), Bursa, Turkey, November 2009».
- [47]. S. Amamra, L. Barazane, M.S. Boucherit and A. Cherifi "Inverse Fuzzy Model Control for a Speed control Induction Motor Based dSPACE Implementation", The 4th International Symposium on Modern Electric Power Systems (MEPS'10), Wroclaw, Poland, September 2010
- [48]. B.K. Bose, "Modern Power Electronics and AC drives", Prentice Hall, Upper Saddle River, 2002.
- [49]. L. Barazane, C. Larbes and M.S. Boucherit.: Robust hybrid control based on feedback linearization and field-oriented controls of induction motor. Archives of Electrical Engineering. Vol. LII, No 2, pp. 115-135, 2003.
- [50]. F. Blaschke, The principle of field orientation applied to the new transvector closed-loop control system for rotating field machines, Siemens-Rev., Vol. 39, pp. 217220, 1972.
- [51]. A.Isidori, "Nonlinear Control Systems - An Introduction", Springer Verlag, Berlin, 1989.
- [52]. J.-S. R. Jang, "Neuro-Fuzzy Modeling: Architectures, Analyses, and Applications", Ph.D. Dissertation, EECS Department, Univ. of California at Berkeley, 1992.
- [53]. J.-S. Roger Jang, C-T. Sun, "Neuro-Fuzzy Modeling and Control", The Proc. of the IEEE, vol. 83, pp. 378-406, Mar. 1995.
- [54]. L.Zadeh, "The concept of a linguistic variable and its application to approximate reasoning", 1975.
- [55]. Dong Hwa Kim GA-PSO based vector control of indirect three phase induction motor", Applied Soft Computing, Elsevier, Volume 7, Issue 2, Pages 601-611, March 2007.
- [56]. M. Ghanai, K. Chafaa, and K. Benmahammed, "Indirect adaptive type-2 fuzzy control for induction motor," Second International Conference on Electrical Systems ICES'06, Oum El Bouaghi University, 2006.
- [57]. N. Goléa, "Identification et commande adaptative floues," Thèse de doctorat d'état en électronique, Université de Batna, 2001.
- [58]. M. Hojati and S. Gazor, "Hybride adaptive fuzzy identification and control of nonlinear systems," IEEE Trans. Fuzzy Syst., vol.10, no. 2, pp. 198-210, Apr. 2002.
- [59]. J. M. Mendel, "Computing Derivatives in Interval Type-2 Fuzzy Logic Systems," IEEE Trans. Fuzzy Syst., vol. 12, no. 1, pp.84-98, Feb. 2004.

- [60]. J. M. Mendel and R. I. B. John, "Type-2 fuzzy sets made simple," IEEE Trans. Fuzzy Syst., vol. 10, no. 2, pp. 117-127, Apr. 2002.
- [61]. C. H. Wang, T. C. Lin, T. T. Lee and H. L. Lin, "Adaptive hybrid intelligent control for uncertain nonlinear dynamical systems," IEEE Trans. Syst., Man, Cybern., Part B, vol. 32, no. 5, pp. 583-597, Oct. 2002.
- [62]. H. Wu and J. M. Mendel, "Uncertainty bounds and their use in the interval type-2 fuzzy logic systems," IEEE Trans. on Fuzzy Syst., vol. 10, no. 5, pp.84-98, Feb. 2004.
- [63]. Mohamed Jemli, Hechmi Ben Azzaa, and Moncef Gossa , "Real-time implementation of IRFOC for Single-Phase Induction Motor drive using dSpace DS 1104 control board", Simulation Modelling Practice and Theory, Elsevier, Volume 17, Issue 6, Pages 1071-1080, July 2009.
- [64]. B. MULTON, S. LEFEBVRE, « MOSFET et IGBT : Circuits de commande », Techniques de l'Ingénieur, réf. D3233, 10 Août 2000.
- [65]. Leroy Somer, 'Machines didactiques de la gamme 1.5 KW Notices techniques', Notices techniques de Leroy Somer, 2004.
- [66]. Control Desk Experiment Guide, dSPACE, May 2002.

ملخص :

أعمال هذا البحث المقدمة في هذه الأطروحة تتمثل خاصة في محورين: تطور الية التحكم في الأنظمة الارتياحية مركزة على تمثيلها العكسي الغامض مطبقة على المحرك الالتزامي و تطوير أساسات التحكم غير منتظمة متركزة على نموذج العكسي الغامض. لقد وجهنا عملنا الى نموذج العكسي الغامض للأنظمة الارتياحية الى هدف الحصول على تقنية جديدة للتحكم المبرين لتدفق الموجة متركز على النظام العكسي الغامض للمحرك الالتزامي. لكي نستفيد أكثر من هذه التقنية الجديدة للنظام العكسي الغامض و توسيع مجال تحمل الارتياحات للمحرك، طرق أخرى للتحكم مثل: معدل اب، الانظمة الانزلاقية و التحكم الملائم كلها متركزة على النموذج العكسي الغامض لقد تطبقت نموذج الغامض تم إنجاز أرضية للتجارب متركزة على بطاقة التحكم dSPACE من أجل إثبات النتائج المتحصل عليها بالمحاكاة

كلمات مفاتيح:

نموذج العلمي الغامض، محرك اللامتزامن، تقنية التدفق الموجة معدل اب، تحكم ذو أساس متحرك غامض متركز على نموذج العكسي الغامض، تحكم ملائم غامض- 2 بطاقة التحكم dSPACE -1130DS.

Résumé:

Les travaux de recherche présentés dans cette thèse portent essentiellement sur deux axes ; Développement d'un algorithme de commande des systèmes incertains basé sur leurs modèles inverse flous appliqué au moteur à induction et Développement des structures de commande non linéaires basées sur le modèle inverse flou. Nous avons orienté notre travail vers la modélisation inverse floue des systèmes incertains dans le but d'élaborer une nouvelle approche de commande robuste à flux orienté basée sur le modèle inverse flou de la machine à induction. Afin de bénéficier de cet avantage apporté par le modèle inverse flou, et d'augmenter la robustesse vis-à-vis de la variation paramétrique de la machine, d'autres techniques de commande à savoir par un régulateur IP anti-saturation, mode glissant ou bien adaptative toutes basées sur la logique floue ont été développées. Une plateforme expérimentale, pilotée par une carte dSPACE DS 1103, a été développée afin d'être utilisée dans les tests de validation expérimentale des différents résultats obtenus en simulation.

Mots clés: Modélisation floue ; Modèle inverse flou; Moteur à induction; Commande vectorielle; Régulateur IP ; Commande à structure variables floue basée sur modèle inverse flou ; Commande adaptative floue-2 ; Carte dSPACE DS1103.

Abstract:

The research works developed in this dissertation are divided in two parts; Development of an algorithm of control of uncertainly systems based with her inverse fuzzy model applied to induction motor and development of non linear control structure based in the inverse fuzzy model. We oriented our work to inverse fuzzy modelisation of uncertainly systems to elaborate a new robust approach of field oriented control based inverse fuzzy model of induction motor. In order to benefit from this advantage giving by the inverse fuzzy model, and overcome all problems of parametric variation of machine, another techniques of control such as IP anti-saturation, sliding control and adaptative all based fuzzy logic are applied.

and to increase the threshold level of the capacitor

A dSPACE based experimental setup was developed to carry out experimental validation tests of the different simulation results.

Keywords: Fuzzy modelisation; Inverse fuzzy model; Induction motor; Vector control; IP regulator; Fuzzy sliding control based inverse fuzzy model; Fuzzy-2 adaptative control; dSPACE DS1103 card.