

Résumé

Le domaine de la variation de vitesse des machines asynchrones s'est développé d'une manière spectaculaire lors des dernières années. Par conséquent, le recours au développement des techniques intelligentes à susciter un intérêt de plus en plus croissant, dans l'industrie, afin de satisfaire des cahiers des charges de plus en plus sévères. Outre les stratégies de commandes conventionnelles (PI, PID), d'autres approches appliquées à la commande des machines asynchrones sont également envisageables : il s'agit de la commande par logique floue et par réseaux de neurones. Le travail présenté dans ce mémoire consiste en la mise au point de ces deux approches intelligentes et à analyser leur performance par rapport au régulateur conventionnel. Tous ces régulateurs de vitesse sont implantés en utilisant la méthode du flux orienté. Les résultats de simulation obtenus sont montrés et interprétés. Nous remarquons que les deux approches présentent une robustesse par rapport aux variations paramétriques de la machine et vis-à-vis des perturbations introduites par le couple résistant.

ملخص

ان تقنية تغيير سرعة المحركات اللاتزامنية تطورت خلال السنوات الاخيرة بصورة كبيرة و هذا من خلال تطور التقنيات الذكية و تطبيقها في الميدان الصناعي على غرار التقنيات العادية نذكر منها التحكم التوافقي التكاملي .بالاضافة الى تقنيات اخرى نخص بالذكر تقنية المنطق الغامض و تقنية الشبكات العصبية العمل المنجز في هذه المذكرة يتضمن وضع نقاط الاساس على هاتين التقنيتين بالمقارنة مع التقنيات العادية ان كل تقنيات التحكم في السرعة منجزة مع طريقة التدفق الموجه كل النتائج المحصل عليها تدل على ان التقنيات المنجزة تتجاوب مع الشروط المفروضة عليها.

Abstract

The field of the variation of the asynchronous machines developed in a spectacular way at the time of the last year. Consequently, the recourse to the development of the intelligent techniques to arouse an interest more and more growing, in industry, in order to satisfy increasingly severe schedules of conditions. In addition to the strategy of conventional orders (PI, PID), of another approaches applied to the ordering of the asynchronous machines are also possible: it is about the order by fuzzy logic and networks of neurons. The work presented to in this memory consists of the development of these two intelligent approaches compared to the regulators conventional. All these speed regulators are established by using the method of directed flow. We notice that the two approaches have a robustness compared to the parametric variations of the machine and the disturbances introduced by the resistive torque.

Table des matières

Introduction générale.....	1
Chapitre I Modélisation de la machine et de son alimentation	
I- Modélisation de la machine asynchrone	3
I.1 Introduction	3
I.2. Présentation de la machine	3
I.3 Modèle transitoire de la machine asynchrone	3
I.3.1 Hypothèses simplificatrices.....	4
I.3.2 Mise en équations de la machine	4
I.3.3 Transformation de Park.....	5
I.3.4 Equations électriques.....	5
I.3.5 Equation magnétiques.....	6
I.3.6 Equation de Park.....	6
I.3.7 Model de Park appliqué aux équations de la machine.....	7
I.4 Choix de la référence.....	8
1.4.1 Référentiel lié au stator.....	9
1.4.2 Référentiel lié au rotor:.....	9
1.4.3 Référentiel lié au champ tournant.....	9
I.5 Modélisation de la machine alimentée en tension	9
I.6 Simulation numérique et interprétation	14
I.7 Modélisation de l'alimentation.....	15
I.7.1 Introduction	15
I.7.2 Modélisation de l'alimentation du MAS à vitesse variable.....	15
I.8 Onduleur de tension triphasé.....	15
I.8.1 Onduleur de tension à <i>MLI</i>	17
I.8.2 Modulation de largeur d'impulsion sinus-triangle.....	17
I.8.3 Simulation et interprétation.....	18
I.9 Conclusion	18

Chapitre II Commande vectorielle

II.1 Introduction 24

II.2 Principe de la commande vectorielle.....24

II.3 Choix du référentiel.....24

 II.3.1 référentiel lié au flux rotorique.....26

II.4 Commande vectorielle26

II.5 Contrôles vectoriel direct et indirect..... 27

 II.5.1 contrôle vectoriel indirect 27

 II.5.2 Bloc de défluxage..... 29

II.6 Régulation, méthodes classiques.....29

 II.6.1 Introduction.....29

 II.6.2 Découplage.....29

 II.6.3 Schéma fonctionnel.....31

II.7 Ajustement des régulateurs32

 II.7.1 Régulateur boucle interne du courant i_{ds} 32

 II.7.2 Réglage de la boucle interne du courant i_{qs} 33

 II.7.3 boucle fermée de régulation de la vitesse.....33

II.8 Contrôle vectoriel direct36

 II.8.1 Estimateur.....35

 II.8.1.1 Calcul de ϕ_r 37

 II.8.1.2 Calcul de ω_s et θ_s 37

 II.8.2 Schéma complet de la commande vectorielle directe à flux rotorique orienté.....37

 II.8.3 Calcul des régulateurs.....39

 II.8.3.1 Régulateur de flux.....39

 II.8.3.2 Régulateur de couple.....40

 II.8.3.3 Régulateur de vitesse41

III.9 simulation et interprétation.....43

III.10 Conclusion.....43

Chapitre III Commande par logique floue

III.1 Principe de la logique floue44

III.2	Domaine d'application	45
III.3	Structure générale d'un système flou.....	45
III.3.1	Eléments de base de la logique floue.....	46
III.3.2	Différentes formes des fonctions d'appartenance.....	47
III.3.3	Intervalles d'inférence	49
III.3.4	Variable linguistique.....	49
III.3.5	Opérateurs de la logique floue.....	50
III.4	Conception d'un contrôleur à logique floue.....	51
III.4.1.	Stratégie de fuzzification.....	51
III.4.2.	base de données	52
III.4.3.	base de règles	52
III.4.4.	Inférence.....	52
III.4.5.	Stratégie de fuzzification	54
III.4.6.	Structure du régulateur par logique floue.....	55
III.5.	Conception d'un contrôleur flou pour le réglage de la vitesse	56
III.5.1.	Le choix des variables d'entrées et de sorties.....	56
III.5.2.	Normalisation des variables d'entrée et de sortie.....	57
III.5.3.	Interface de fuzzification.....	57
III.5.4.	Inférence.....	58
III.5.5.	Traitement numérique de l'inférence	59
III.5.6.	Défuzzification.....	59
III.5.6.1	Etude des différents sous ensembles	59
III.6.	Simulation et interprétation.....	61
III.7.	Conclusion.....	61

Chapitre IV Synthèse de régulateur neurone

IV.1	Historique des réseaux de neurones.....	69
IV.2.	Application.....	70
IV.3.	Définition du neurone formel.....	70
IV.4.	Elément de base des réseaux de neurones.....	70
IV.5.	Fonctions de transfert.....	72
IV.6.	Réseaux de neurone	73

IV.6.1. Choix du nombre d'unités cachées.....	75
IV.6.2 Propriétés de réseaux de neurones	75
IV.6.3 Réseaux non bouclés et réseaux bouclés.....	77
IV.6.3.1 Les réseaux non bouclés.....	77
IV.6.3.2. Les réseaux bouclés.....	77
IV.6.4. Réseaux non bouclés complètement connectés et réseaux à couches.....	78
IV.6.4.1. Les réseaux non bouclés complètement connectés.....	78
IV.6.4.2. Les réseaux non bouclés à couches.....	78
IV.7. Processus d'apprentissage.....	79
IV.7.1. L'apprentissage de Widrow-Hoff	81
IV.8. Les différents types de réseaux de neurones	82
IV.9 Le réseau de rétro propagation.....	82
IV.10 Régulation de la vitesse par des réseaux de neurones.....	84
IV.10.1 Structure du réseau de neurone.....	84
IV.11 Simulation et interprétation	85
IV.12 Conclusion.....	86
Conclusion générale	94
Annexe.....	95
Bibliographie	96

Listes des tableaux

Tab	Titre	N
Tab III.1	Symbole de cinq intervalles ou sous ensembles flous	49
Tab III.2	Matrice d'inférence de 7 ensembles flous	59
Tab III.3	Matrice d'inférence de 3 ensembles flous	60
Tab III.4	Matrice d'inférence de 5 ensembles flous	60
Tab IV.1	fonction de transfert $a=f(n)$	73

Liste des figures

Figure 1 représentation des enroulements statoriques et rotorique

figure.1.1 simulation du démarrage sous pleine tension à vide

Figure. 1.2 *Démarrage sous pleine tension avec application d'un couple Résistant ($c_r = 5$ N.m) à l'instant $t=1s$*

Figure 2 Schéma d'un onduleur de tension alimentant le stator du MAS

Figure. 2.1 MLI sinus –triangle bipolaire

Figure. 2. 3 Association convertisseur – machine à vide

Figure.2.4 *Association convertisseur – machine avec une charge de 5 Nm à l'instant $t=1 s$*

Figure.2.5 Association convertisseur – machine avec application d'une charge à $t=0.5s$ et $t=0.8s$ et inversement de sens de rotation

Figure II.1 Repère d q lié au flux rotorique

Figure. II.2 Structure du bloc F.O.C

Figure II.3 reconstitution des tensions v_{ds} et v_{qs}

Figure II.4 commande découplée – expression de ϕ_r et c_e

Figure. II.5. Schéma bloc de la commande vectorielle indirecte.

Figure. II.6 schéma fonctionnel de régulation de courant i_{ds} .

Figure. II.7 schéma fonctionnel de régulation de vitesse.

Figure II.8 commande vectorielle directe de flux d'une machine alimentée en tension

Figure II.9 schéma en boucle ouverte

Figure II.10 schéma en boucle ouverte

Figure II.11 schéma en boucle fermée

Figure. III.1 *schéma générale d'un bloc fonctionnel flou.*

Figure. III.2 Niveaux d'appartenance en logique booléenne

Figure. III.3 Niveaux d'appartenance en logique floue

Figure III.5 Représentation des variables linguistiques

Figure III.6 Univers de discours normalisé dans l'intervalle $[-1 +1]$ avec une partition de 3, 5,7 ensembles flous

Figure III.7 Exemple d'inférence à plusieurs règles

Figure.III.8 Structure d'un régulateur à logique floue (RLF)

Figure III.9 Description des fonctions d'appartenance

Figure III.10 Vitesse et couple de 3 sous ensembles

Figure III.11 Vitesse et couple de 5 sous ensembles

Figure III.12 Vitesse et couple de 7 sous ensembles

Figure III.13 Comparaison des sous ensembles 3,5 et 7

Figure III.11 Réglage par logique floue de la vitesse lors d'un démarrage à vide pour une vitesse de 100 rd/s

Figure III.12 Réglage par logique floue de la vitesse avec application d'un couple de charge à $t=0.5$ s

Figure IV.1 le neurone artificiel

Figure IV.2 *Fonctions de transfert*

Figure IV.3 topologie des réseaux de neurone

Figure IV.4 Réseau de neurone complètement connecté

Figure IV.5 réseau de neurone avec une couche de neurone cachés et un neurone de sortie linéaire

Figure IV.6 Réseau de rétro propagation.

Figure IV.7 structure de réseau neuronal utilisé pour la régulation de vitesse

Fig.IV.8 vitesse et couple de 3neurones

Fig. IV.9 Vitesse et couple de 5neurones

Fig. IV.10 Vitesse et couple de 7neurones

Fig. IV.11 Comparaison des vitesses de régulateur neurone

Fig.IV.12 Réglage par réseau de neurone de la vitesse lors d'un démarrage à une Vitesse de 100 rd/s

Fig.IV.13 Réglage par réseau de neurone de la vitesse avec application d'un couple de charge de 5 n.m à $t=0.5$ s

Fig. IV.14 Réglage par réseau de neurone de la vitesse lors d'un démarrage à une Vitesse de 100 rd/s et inversement de sens de rotation à $t=0.5$ s

Introduction générale

A l'heure actuelle, la machine asynchrone détrône la machine à courant continu dans le domaine de la variation de vitesse. En effet, de part sa robustesse, l'absence d'un collecteur mécanique et de son faible coût, elle s'avère une solution adéquate pour la plupart des industrielles. En outre, les stratégies de contrôle et de commande dont elle peut être dotée lui procurent des comportements et des performances équivalents à ceux de la machine à courant continu.

Le développement de l'électronique de puissance et de la micro - informatique ont contribué à créer chez plusieurs équipes de recherche un intérêt, de plus en plus croissant, pour les techniques numériques de commande et de régulation, qui appliquées à la machine asynchrone, sont amenées à satisfaire des cahiers de charges de plus en plus sévères. Parmi les techniques de commande utilisées : partant d'une connaissance de la machine et de l'environnement dans lequel elle est destinée à être utilisée. Les régulateurs classiques (PI, PID...) ont donc besoin d'un modèle de la machine ; ce qui ne se fait pas dans certains cas d'une manière aisée. De nouvelles approches ont suscité récemment l'attention dans le domaine de la commande des machines. Il s'agit de l'application de la logique floue et des réseaux de neurones.

les concepts de ces deux approches sont basés sur la perception globale qu'un être humain à l'environnement qui l'avoisine pour la logique floue, et pour les réseaux de neurones, sur une tentative de modéliser le cerveau humain afin de pouvoir explorer ses capacités, de ce fait, ces deux approches, qualifiées alors d'intelligentes font preuve de leur efficacité dans divers domaines d'application (traitement d'image, vision par ordinateurs, reconnaissance de la parole et de caractères manuscrits) . Par ailleurs, le domaine de la commande des systèmes, et plus particulièrement des entraînements électriques, s'avère un champ d'application intéressant. Le travail présenté dans ce mémoire consiste à étudier ces deux approches et à les mettre en œuvre pour la commande de la vitesse d'une machine asynchrone, et ce dans le but de comparer leurs performances avec celles utilisant des techniques conventionnelles. Ce mémoire est présenté en quatre chapitres. Dans le premier, on présente la modélisation de la machine et puis celle de son alimentation par un onduleur de tension. Le deuxième chapitre, concerne l'application de la commande vectorielle à la machine asynchrone de but de la commande comme une machine à courant

continu, dans le troisième chapitre, nous détaillerons les formalismes de base de la logique floue, puis la méthodologie adoptée pour la conception du système de commande basé sur cette approche. Lors du dernier chapitre, nous nous intéresserons tout d'abord à une présentation générale des réseaux de neurones formels, ensuite à la synthèse de correcteurs utilisant ces structures neurométriques. Enfin, la dernière partie consistera à une étude comparative des performances des systèmes étudiés. La comparaison des résultats de simulation porteront, en fait, sur tout le domaine de fonctionnement de la machine asynchrone, en concerneront son comportement aussi bien vis à vis des défauts de paramétrage que face à l'introduction d'une charge mécanique non linéaire.

Chapitre I

***Modélisation de la machine et
de son alimentation***

MODELISATION DE LA MACHINE ASYNCHRONE ET DE SON ALIMENTATION

I.1 Introduction

Actuellement le moteur asynchrone est de plus en plus utilisé pour assurer un fonctionnement à vitesse ou positionnement variables et ceci pour plusieurs raisons :

- C'est le moteur le plus robuste et le moins cher sur le marché.
- Sa commande, assez complexe, peut être réalisée sans trop de difficulté grâce aux progrès de la commande numérique.
- Le développement des semi- conducteurs commandés à l'ouverture autant qu'à la fermeture (GTO, IGBT, ... selon le niveau de puissance) permet l'utilisation de ce moteur dans une large gamme de vitesse et de puissance et en particulier au-delà des limites en puissance imposées, autrefois, par la commutation forcée.

L'inconvénient principale du moteur asynchrone est qu'il possède une puissance massique plus faible que celle des moteurs synchrones à aimants et qu'il a des pertes au rotor qui réduisent son rendement et provoquent un échauffement gênant surtout pour les basses vitesses. Un autre inconvénient réside dans le couplage du flux par rapport au couple électromagnétique.

I.2 Présentation de la machine

Une machine asynchrone à cage est constituée de deux parties, *le stator* et *le rotor*. *Le stator* est la partie statique de la machine. Il est constitué d'un circuit magnétique comportant de multiples encoches à l'intérieur desquelles sont bobinées trois enroulements formant les enroulements statoriques. Au centre de ce cylindre, on retrouve le *rotor* de la machine dont le circuit magnétique est composé de barres généralement en cuivre ou en aluminium coulé. Ces barres sont reliées entre elles à chaque extrémité par un anneau de court-circuit. Cette cage est traversée par l'arbre mécanique qui peut sortir ou non de chaque côté [1].

I.3 Modèle transitoire de la machine asynchrone

Il nous faut un modèle de la machine asynchrone qui permet de simuler le fonctionnement en régime transitoire et d'appliquer un contrôle vectoriel indirect par orientation du flux rotorique.

I.3.1 Hypothèses simplificatrices

Du fait que la machine est un système complexe et non linéaire, plusieurs phénomènes sont introduits lors du fonctionnement. Ce qui rend la machine difficile à commander et à modéliser. Pour palier à ce problème, on tient compte des hypothèses simplificatrices suivantes [2]:

- Le circuit magnétique de la machine n'est pas saturé ;
- On néglige l'effet des pertes fer ;
- La répartition de l'induction magnétique dans l'espace est sinusoïdale, et la perméabilité est constante ;
- Les effets des courants de Foucault, ainsi que l'effet de peau et d'échauffement sont négligés ;
- L'entrefer de la machine est constant et l'effet des encoches est négligé ;
- Les inductances propres sont des constantes et les inductances mutuelles ne dépendent que de la position des enroulements.

1.3.2 Mise en équations de la machine

Les enroulements des trois phases statoriques et des trois phases rotoriques dans l'espace peuvent être représentés comme indiqué en figure I.1. Les phases rotoriques sont court-circuitées sur elles mêmes.

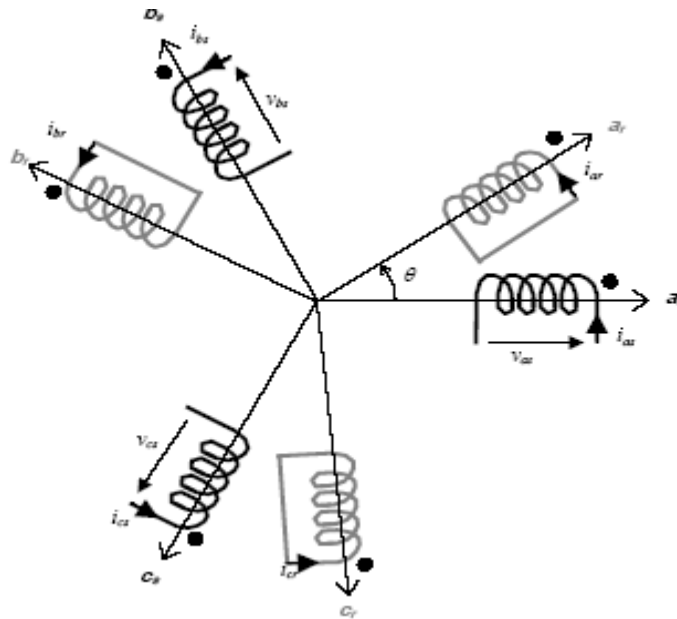


Fig. I.1 représentation des enroulements statoriques et rotoriques

I.3.3 Transformation de Park

La transformation de PARK consiste à appliquer aux courants, tensions et flux, un changement de variable en faisant intervenir l'angle entre les axes des enroulements réels, d'enroulements fictifs d_s , q_s , d_r , q_r , dont les axes magnétiques sont liés aux axes d et q .

Donc, en utilisant la transformée de Park, on transforme le modèle de la machine asynchrone d'un modèle triphasé à un modèle biphasé (d : axe direct et q : axe quadrature).

$$P(\theta) = \left(\frac{2}{3} \right) \begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\ -\sin(\theta) & -\sin(\theta - \frac{2\pi}{3}) & -\sin(\theta + \frac{2\pi}{3}) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad (\text{I.1})$$

$$P^{-1}(\theta) = \left(\frac{2}{3} \right) \begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \frac{1}{2} \\ \cos(\theta - \frac{2\pi}{3}) & -\sin(\theta - \frac{2\pi}{3}) & \frac{1}{2} \\ \cos(\theta + \frac{2\pi}{3}) & -\sin(\theta + \frac{2\pi}{3}) & \frac{1}{2} \end{bmatrix} \quad (\text{I.2})$$

I.3.4 Equations électriques

Statorique:

$$\begin{bmatrix} V_{sa} \\ V_{sb} \\ V_{sc} \end{bmatrix} = R_s \begin{bmatrix} i_{sa} \\ i_{sb} \\ i_{sc} \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} \Phi_{sa} \\ \Phi_{sb} \\ \Phi_{sc} \end{bmatrix} \quad (\text{I.3})$$

rotorique

$$\begin{bmatrix} V_{ra} \\ V_{rb} \\ V_{rc} \end{bmatrix} = R_r \begin{bmatrix} i_{ra} \\ i_{rb} \\ i_{rc} \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} \Phi_{ar} \\ \Phi_{br} \\ \Phi_{cr} \end{bmatrix} \quad (\text{I.4})$$

I.3.5 Equations magnétiques

Une matrice des inductances $[L(\theta)]$ établit la relation entre le s flux et les courants; elle comporte 36 coefficients non nuls dont la moitié dépend du temps. En fonction de la position du rotor (θ);

Soit:

$$\begin{bmatrix} \Phi_{s_i} \\ \Phi_{r_i} \end{bmatrix} = \begin{bmatrix} [L_s] & [M_{sr}] \\ [M_{rs}] & [L_r] \end{bmatrix} \times \begin{bmatrix} i_{s_i} \\ i_{r_i} \end{bmatrix} \quad \text{ou } i=a,b,c \quad (\text{I.5})$$

Avec:

$$[L_s] = \begin{bmatrix} L_s & M_s & M_s \\ M_s & L_s & M_s \\ M_s & M_s & L_s \end{bmatrix} \quad (\text{I.6})$$

Et $M_{sr} = M_{rs} = M$

Finalement:

$$V_{si} = [R_s][i_{si}] + \frac{d([L_{sr}][i_{si}] + [M_{sr}][i_{ri}])}{dt} \quad (\text{I.7})$$

$$V_{ri} = [R_r][i_{ri}] + \frac{d([M_{sr}][i_{si}] + [L_r][i_{ri}])}{dt}$$

$$[L_r] = \begin{bmatrix} L_r & M_r & M_r \\ M_r & L_r & M_r \\ M_r & M_r & L_r \end{bmatrix} \quad (\text{I.8})$$

$$[M_{sr}] = [M_{rs}]^T = M_{SR} \begin{bmatrix} \cos(\theta) & \cos(\theta + \frac{2\pi}{3}) & \cos(\theta - \frac{2\pi}{3}) \\ \cos(\theta - \frac{2\pi}{3}) & \cos(\theta) & \cos(\theta + \frac{2\pi}{3}) \\ \cos(\theta + \frac{2\pi}{3}) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta) \end{bmatrix} \quad (\text{I.9})$$

I.3.6 Equations de Park

La tension aux bornes d'un enroulement est la somme de la force électromotrice (fem) due à la variation du flux, et de la chute de tension Ohmique. En adoptant la convention récepteur le système d'équation est:

$$\begin{cases} V_a = R.i_a + \frac{d\Phi_a}{dt} \\ V_b = R.i_b + \frac{d\Phi_b}{dt} \\ V_c = R.i_c + \frac{d\Phi_c}{dt} \end{cases} \quad (\text{I.10})$$

I.3.7 Model de Park appliqué aux équations de la machine

On applique une transformation de PARK d'angle θ_s aux variables statoriques et d'angle θ_r aux variables rotoriques.

$$[V_{s_{d,q,o}}] = R[is_{d,q,o}] + \frac{d}{dt}[\Phi_{s_{d,q,o}}] + \left([P^{-1}(\theta_s)] \frac{d}{dt} [P(\theta_s)] \right) (\Phi_{s_{d,q,o}}) \quad (\text{I.11})$$

$$[V_{r_{d,q,o}}] = R[ir_{d,q,o}] + \frac{d}{dt}[\Phi_{r_{d,q,o}}] + \left([P^{-1}(\theta_r)] \frac{d}{dt} [P(\theta_r)] \right) (\Phi_{r_{d,q,o}})$$

$$[\Phi_{s_{d,q,o}}] = [P^{-1}(\theta_s)][L_{ss}][P(\theta_s)][is_{d,q,o}] + [P^{-1}(\theta_s)][M_{sr}]^t [P(\theta_r)][ir_{d,q,o}] \quad (\text{I.12})$$

$$[\Phi_{r_{d,q,o}}] = [P^{-1}(\theta_r)][M_{sr}]^t [P(\theta_s)][is_{d,q,o}] + [P^{-1}(\theta_r)][L_{rr}][P(\theta_r)][ir_{d,q,o}]$$

Après calcul:

$$[P^{-1}(\theta_s)][L_{ss}][P(\theta_s)] = \begin{bmatrix} L_s + 2M_s & 0 & 0 \\ 0 & L_s - M_s & 0 \\ 0 & 0 & L_s - M_s \end{bmatrix} = \begin{bmatrix} L_{so} & 0 & 0 \\ 0 & L_{so} & 0 \\ 0 & 0 & L_{so} \end{bmatrix} \quad (\text{I.13})$$

$$[P^{-1}(\theta_r)][L_{rr}][P(\theta_r)] = \begin{bmatrix} L_r + 2M_r & 0 & 0 \\ 0 & L_r - M_r & 0 \\ 0 & 0 & L_r - M_r \end{bmatrix} = \begin{bmatrix} L_r & 0 & 0 \\ 0 & L_r & 0 \\ 0 & 0 & L_r \end{bmatrix} \quad (\text{I.14})$$

$$[P^{-1}(\theta_s)][M_{sr}][P(\theta_r)] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{3}{2}M\cos(\theta_s - \theta_r - \theta) & -\frac{3}{2}M\sin(\theta_s - \theta_r - \theta) \\ 0 & \frac{3}{2}M\sin(\theta_s - \theta_r - \theta) & \frac{3}{2}M\cos(\theta_s - \theta_r - \theta) \end{bmatrix} \quad (\text{I.15})$$

Généralement, on choisit le même référentiel (d,q) pour les variables statoriques et rotoriques .
Alors, les équations du modèle s'écrivent :

$$\begin{aligned} V_{so} &= R_s i_{so} + \frac{d}{dt} \Phi_{so} \\ V_{sd} &= R_s i_{sd} + \frac{d}{dt} \Phi_{sd} - \frac{d\theta_s}{dt} \Phi_{sq} \\ V_{sq} &= R_s i_{sq} + \frac{d}{dt} \Phi_{sq} + \frac{d\theta_s}{dt} \Phi_{sd} \end{aligned} \quad (\text{I.16})$$

$$V_{ro} = R_r i_{ro} + \frac{d}{dt} \Phi_{ro}$$

$$V_{rd} = R_r i_{rd} + \frac{d}{dt} \Phi_{rd} - \frac{d\theta_r}{dt} \Phi_{rq}$$

$$V_{rq} = R_r i_{rq} + \frac{d}{dt} \Phi_{rq} + \frac{d\theta_r}{dt} \Phi_{rd}$$

I.4 Choix de la référence

L'utilisation de la transformation de PARK, nécessite le choix d'un référentiel qui permet de simplifier au maximum les expressions analytiques du modèle.

I.4.1 Référentiel lié au stator

La position des axes d,q par rapport aux axes statoriques et rotoriques est définie par les relations suivantes:

$$\frac{d\theta_s}{dt} = 0; \quad \frac{d\theta_r}{dt} = -w_r \quad (\text{I.17})$$

Ce référentiel est choisi dans le cas de variations importantes de la vitesse de rotation associées ou non à des variations de la fréquence d'alimentation.

I.4.2 Référentiel lié au rotor

La position des axes d, q par rapport aux axes statoriques et rotoriques est la suivante:

$$\frac{d\theta_s}{dt} = w_r; \quad \text{et} \quad \frac{d\theta_r}{dt} = 0 \quad (\text{I.18})$$

Ce choix est nécessaire dans l'étude des régimes transitoires où la vitesse de rotation est constante.

I.4.3 Référentiel lié au champ tournant

Ce type de référentiel est caractérisé par les relations suivantes :

$$\frac{d\theta_s}{dt} = w_s; \quad \frac{d\theta_r}{dt} = w_s - w_r \quad (\text{I.19})$$

On choisie ce référentiel lorsqu'on veut étudier les problèmes dont la fréquence d'alimentation est constante.

I.5. Modélisation de la machine alimentée en tension

Afin de réduire le nombre d'équations, nous utilisons la représentation biphasée, pour une meilleure représentation de la machine asynchrone, nous prenons comme variable d'état i_{sd} , i_{sq} , ϕ_{rd} , ϕ_{rq} et ω_m . Le modèle de la machine dans un repère (d-q) lié au champ tournant à la vitesse de synchronisme est:

$$\frac{di_{sd}}{dt} = \frac{1}{\sigma L_s} \left[-R_s i_{sd} + \sigma \omega_s L_s i_{sq} - \frac{l_m}{l_r} \frac{d\Phi_{rd}}{dt} + \omega_s \frac{L_m}{L_r} \Phi_{rq} + V_{sd} \right] \quad (\text{I.20})$$

$$\frac{di_{sq}}{dt} = \frac{1}{\sigma L_s} \left[-\sigma \omega_s L_s i_{sd} - R_s i_{sq} - \omega_s \frac{L_m}{L_r} \Phi_{rd} - \frac{L_m}{L_r} \frac{d\Phi_{rq}}{dt} + V_{sq} \right] \quad (\text{I.21})$$

$$\frac{d\Phi_{dr}}{dt} = \frac{L_m}{T_r} i_{ds} - \frac{1}{T_r} \Phi_{dr} + \omega_{gl} \Phi_{qr} \quad (\text{I.22})$$

$$\frac{d\Phi_{qr}}{dt} = \frac{L_m}{T_r} i_{qs} - \omega_{gl} \Phi_{dr} - \frac{1}{T_r} \Phi_{qr} \quad (\text{I.23})$$

$$\frac{d\omega_m}{dt} = \frac{p}{J} (C_e - C_r) \quad (\text{I.24})$$

$$C_e = \frac{pL_m}{L_r} (\Phi_{rd} i_{sq} - \Phi_{rq} i_{sd}) \quad (\text{I.25})$$

$$\omega_{gl} = \omega_s - \omega_m$$

$$\sigma = 1 - \frac{L_m^2}{L_r L_s} \quad (\text{I.26})$$

σ : Coefficient de dispersion

Nous calculons les flux statoriques (ϕ_{sd} , ϕ_{sq}) et les courants rotoriques (i_{sd} , i_{sq}) par les relations suivantes:

$$\begin{aligned} \Phi_{sd} &= \frac{L_m}{L_r} \Phi_{rd} + \sigma L_s I_{sd} \\ \Phi_{sq} &= \frac{L_m}{L_r} \Phi_{rq} + \sigma L_s I_{sq} \end{aligned} \quad (\text{I.27})$$

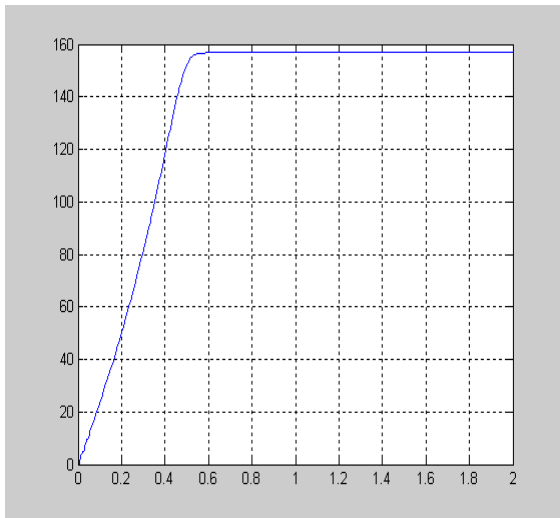
$$I_{rq} = \frac{1}{L_r} (\Phi_{rq} - L_m I_{sq})$$

$$I_{rd} = \frac{1}{L_r}(\Phi_{rd} - L_m I_{sd})$$

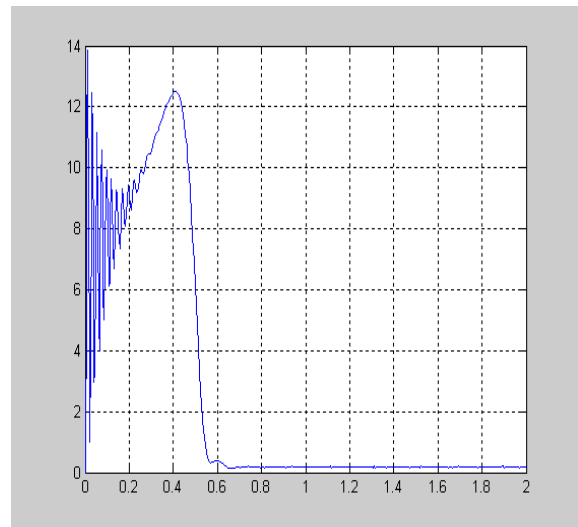
Le flux rotorique (ϕ_r) et le courant dans une phase (i_{sa}) peuvent être calculer à partir des grandeurs (ϕ_{rd}, ϕ_{rq}) et (i_{sd}, i_{sq}), selon les expressions suivantes :

$$\Phi_r = \sqrt{\Phi_{rd}^2 + \Phi_{rq}^2}$$

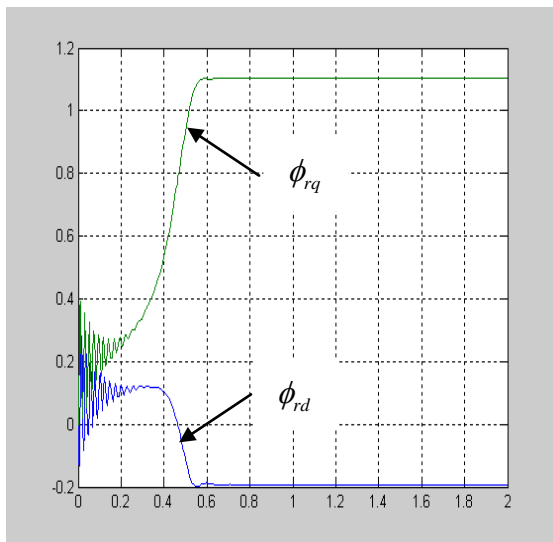
$$I_{as} = \sqrt{\frac{2}{3}}(i_{sd} \cos(\theta_s) + i_{sq} \sin(\theta_s)) \quad (\text{I.28})$$



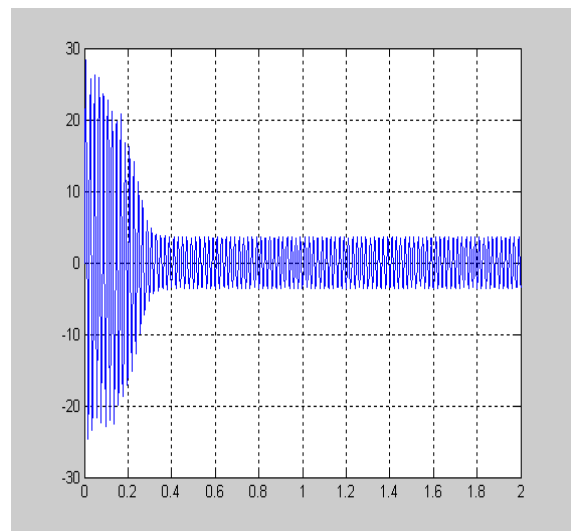
a. La vitesse de rotation (rd/s)



b- couple électromagnétique (N.m)

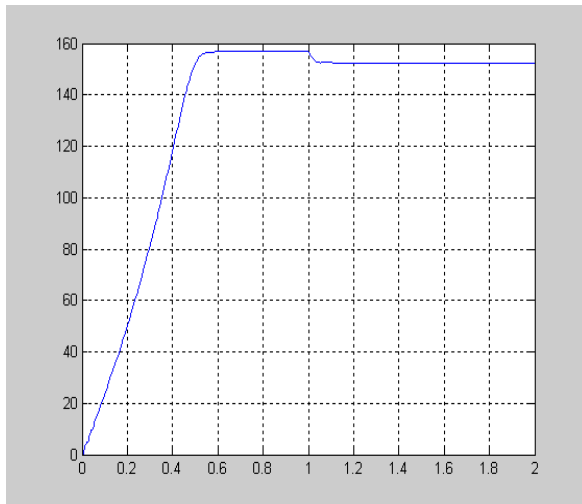


c- les flux dr-qr (wb)

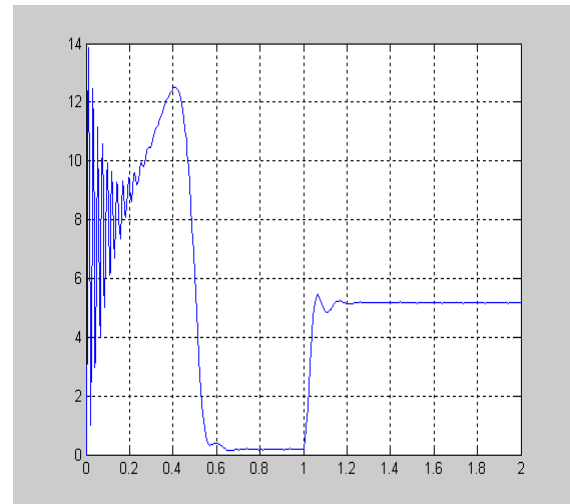


d- courant I_{as} (A)

fig.I.2 simulation du démarrage sous pleine tension à vide



a. La vitesse de rotation (rd/s)



b- couple électromagnétique (N.m)

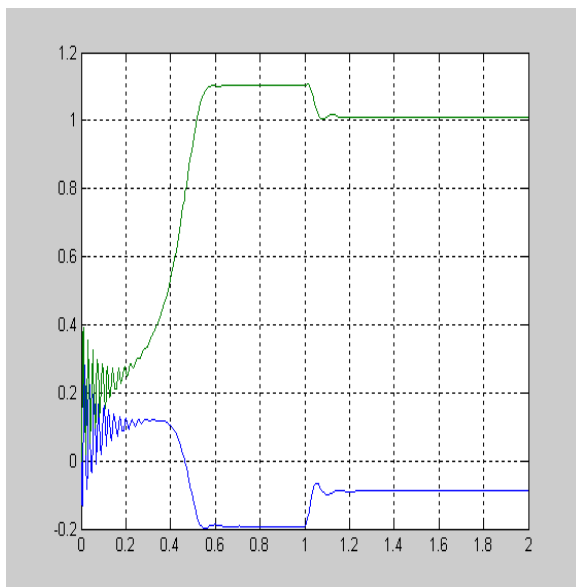
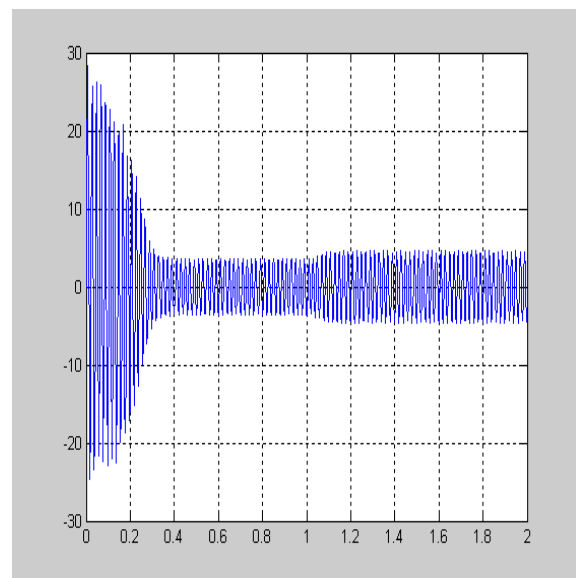
c- les flux d_r - q_r (wb)d- courant I_{as} (A)

Fig. I.3 Démarrage sous pleine tension avec application d'un couple Résistant ($c_r = 5 \text{ N.m}$) à l'instant $t=1\text{s}$

I.6 Simulation numérique et interprétation

Le but est de voir le comportement dynamique de la machine à travers les différentes variables.

Les résultats de simulation sont présentés sur les figures (1.2) et (1.3). Ces figures illustrent respectivement l'évolution de la vitesse de rotation, le couple électromagnétique et le flux magnétique selon l'axe d et q.

Ces résultats sont obtenus lors d'un démarrage sous pleine tension de la machine avec et sans application d'un couple de charge. On constate que la vitesse atteint sa valeur nominale $\Omega_n = 157$ rd/s dans un temps $t \approx 0.5$ s. lorsque cette vitesse se stabilise, le couple électromagnétique tend vers zéro, à cause de leur couplage avec le flux rotorique, on constate aussi la chute de la vitesse à cause de l'augmentation de la charge.

Il est aussi important de souligner que la variation du couple. Est suivie de la variation des deux composantes du flux, ce qui montre le couplage entre le flux et le couple. Alors, afin de rendre une commande analogue à celle d'une machine à courant continu. Ceci sera détaillé dans le chapitre suivant.

I.7 Modélisation de l'alimentation

I.7.1 Introduction

L'augmentation toujours croissante des performances globales des entraînements industriels à vitesse variable, est aujourd'hui principalement liée aux progrès réalisés dans le domaine de la commande des machines à courant alternatif. Ceci grâce au développement de la technologie des composants de l'électronique de puissance à fréquence élevée et à forte puissance de calcul.

I.7.2 Modélisation de l'alimentation du MAS à vitesse variable

L'alimentation doit pouvoir fournir et récupérer en cas de freinage, l'énergie électrique du moteur. Elle est généralement composée d'un convertisseur statique alimenté par une source de tension continue constante, qui permet d'imposer aux enroulements statiques de la machine des tensions d'amplitude et de fréquence réglables en agissant sur la commande des semi-conducteur du convertisseur statique, elle présente la solution adéquate pour le démarrage des moteurs asynchrones.

I.8 Onduleur de tension triphasé

Un onduleur de tension triphasé dont les composants semi-conducteurs contrôlables sont les transistors ou des thyristors GTO, est le cœur de l'organe de commande de MAS. Il peut être considéré comme un amplificateur de puissance, et il est constitué de trois bras, ayant deux interrupteurs pour chacun. Chaque interrupteur est monté en parallèle inverse avec une diode de récupération. Pour assurer la continuité des courants alternatifs et éviter le court-circuit de la source, les interrupteurs $K_i (i=1..6)$ doivent être contrôlés de manière complémentaire. Le schéma structurel d'un tel convertisseur statique alimentant le stator du MAS est illustré par la figure suivante :

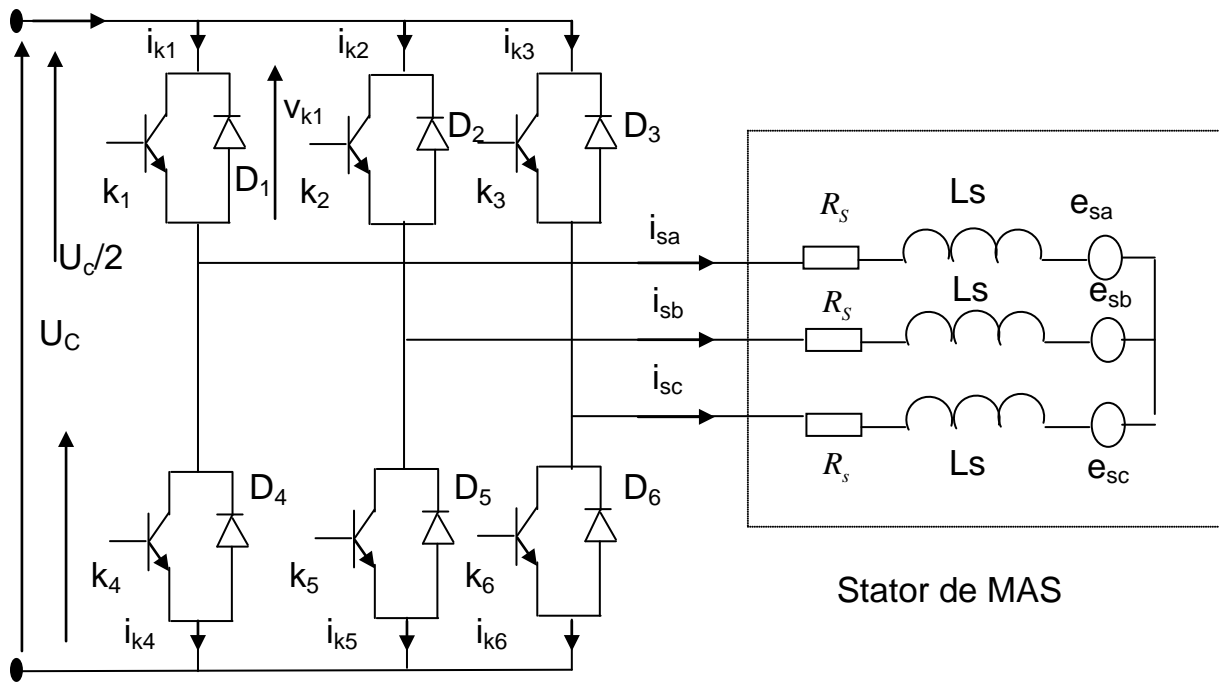


Fig. I.4 Schéma d'un onduleur de tension alimentant le stator du MAS

Pour simplifier l'étude, on supposera que :

- La commutation des interrupteurs est instantanée ;
- La chute de tension aux bornes des interrupteurs est négligeable ;
- La charge triphasée est équilibrée, couplée en étoile avec un neutre isolé.

Les tensions simples s'écrivent en fonction des tensions composées U_{sab} , U_{sbc} , U_{sca} , sous la forme suivante :

$$V_{sa} = \frac{1}{3} (U_{sab} - U_{sca}) \quad (I.29)$$

$$V_{sb} = \frac{1}{3} (U_{sbc} - U_{sab}) \quad (I.30)$$

$$V_{sc} = \frac{1}{3} (U_{sca} - U_{sbc}) \quad (I.31)$$

Les diodes $D_{i=1,2,\dots,6}$ sont des diodes de protection des transistors assurant la roue libre ou la récupération. Plusieurs méthodes sont utilisées pour commander les interrupteurs d'un onduleur, parmi les stratégies la plus utilisée est la Modélisation de Largeur d'Impulsion (MLI).

I.8.1 Onduleur de tension à *MLI*

Plusieurs types de *MLI* se distinguent, dont on peut citer [3] [4] :

- les modulations *sinus - triangle* effectuant la comparaison d'un signal de référence sinusoïdale à une porteuse en généralement triangulaire.
- Les modulations *précalculées* pour lesquelles les angles de commutations sont calculés hors ligne pour annuler certaines composantes du spectre de la tension, et donner une onde fondamentale.
- Les modulations *postcalculées* appelées encore *MLI* régulières symétrique ou *MLI* vectorielles dans lesquelles les angles de commutation sont calculés on ligne.

I.8.2 Modulation de largeur d'impulsion *sinus-triangle*

La *MLI* permet de former à la sortie d'un onduleur chaque alternance de la tension à l'aide d'une succession de créneaux rectangulaires, dont la largeur varie suivant une loi de commande qui a pour but :

- repousser les fréquences des harmoniques pour faciliter leur filtrage.
- faire varier la valeur efficace du fondamentale de la tension de sortie.

Le réglage est effectué par les durées d'ouverture et de fermeture des interrupteurs et par les séquences de fonctionnement. La lois de modulation est une comparaison entre une tension de référence sinusoïdale $V_{ref}(t)$ et une onde triangulaire appelée porteuse $V_p(t)$ de fréquence (fp) [3] [4], le signal de commande de l'interrupteur électronique d'un bras de l'onduleur triphasé $k_{i=1,2,3}$ est fermé si l'onde de référence de la phase correspondante est supérieure ou égale à la porteuse triangulaire, et vis versa [5].

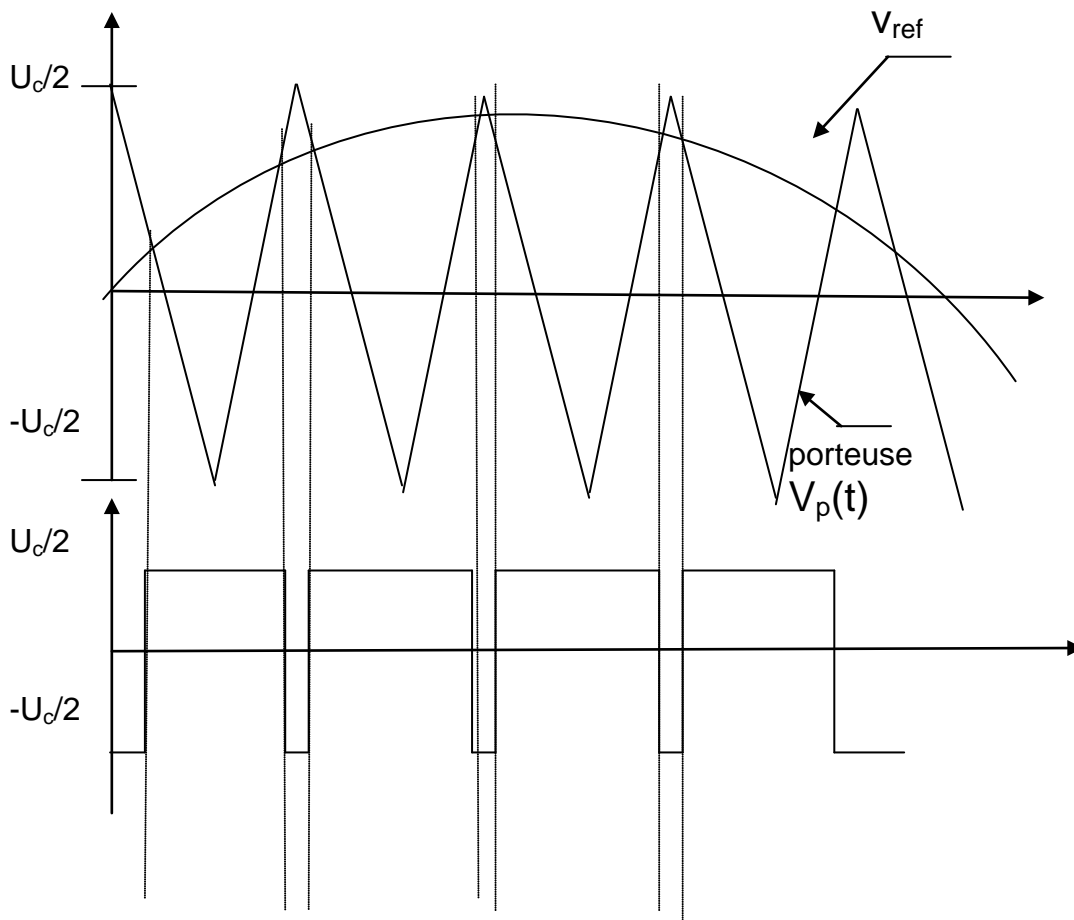


Fig. I.5 MLI sinus –triangle bipolaire

I.8.3 Simulation et interprétation

Les résultats de simulation des figures (I.7 , I.8, I.9) illustrent l'évolution de la vitesse de rotation, du couple électromagnétique, des flux rotoriques (ϕ_{dr}, ϕ_{qr}) et de courant de phase.

On présente d'abord l'association machine convertisseur à vide, puis application d'une charge à $t=1s$, finalement l'inversion de sens de rotation, la vitesse se stabilise à la valeur de 157 rd/s à $t=0.2 s$ et le couple s'annule à cette valeur . Le couple électromagnétique suit le couple de charge.

I.9 Conclusion

On a représenté dans ce chapitre la machine asynchrone, sa modélisation, on mettant en exergue la complexité et la non linéarité du modèle. Par la suite, et en se basant sur un ensemble d'hypothèses simplificatrices, le modèle de MAS dans le repère de PARK a été établi dans le but de linéariser le système et faciliter l'étude. Puis on a abordé la modélisation de la partie d'alimentation, le principe de fonctionnement et de commande de l'onduleur de

tension triphasé a été présenté en donnant les principes de *MLI* les plus connus. Sur la base du modèle retenu de la machine un programme établi sur le logiciel *MATLAB* (simulink), a permis de simuler le comportement dynamique de la machine grâce à lequel on a représenté les résultats précédents.

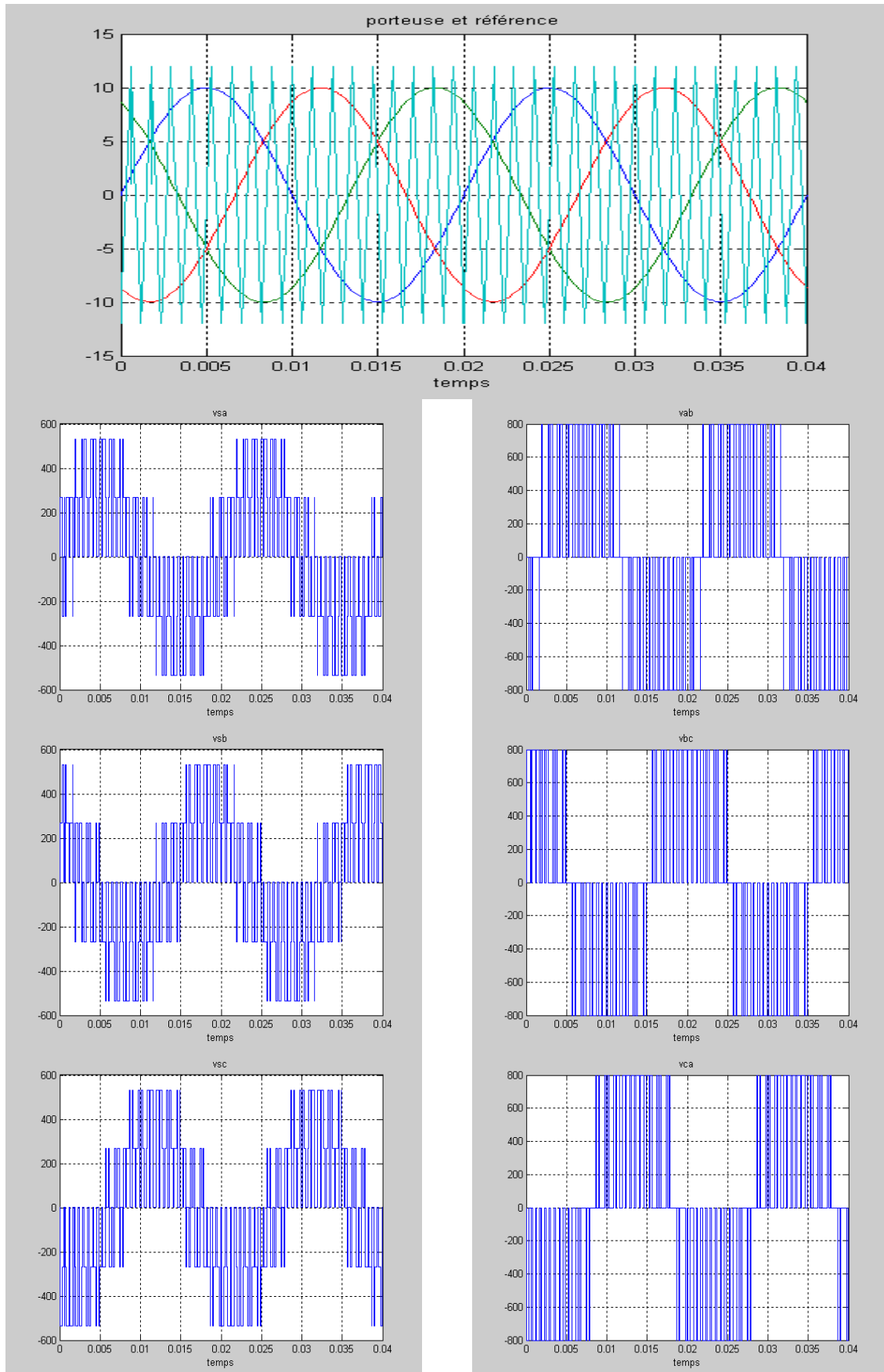
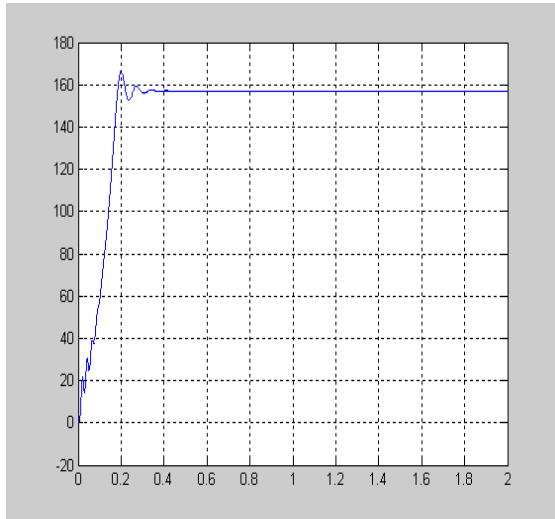
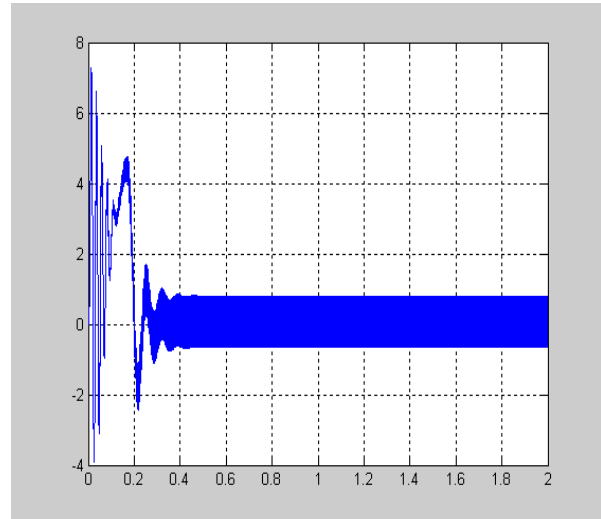


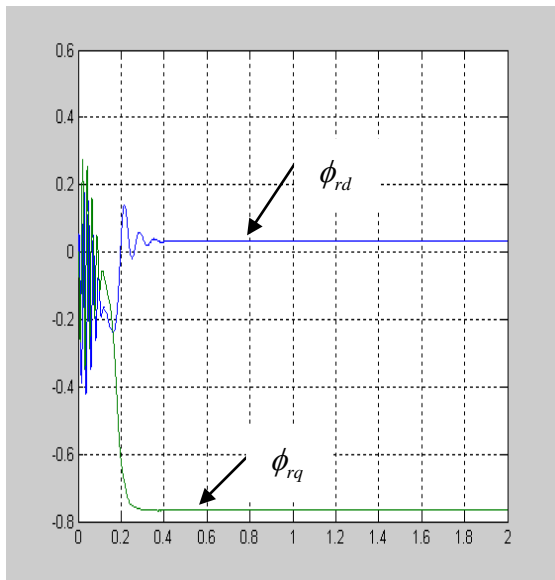
Fig.1.6 Signaux de commande et de sortie de l'onduleur



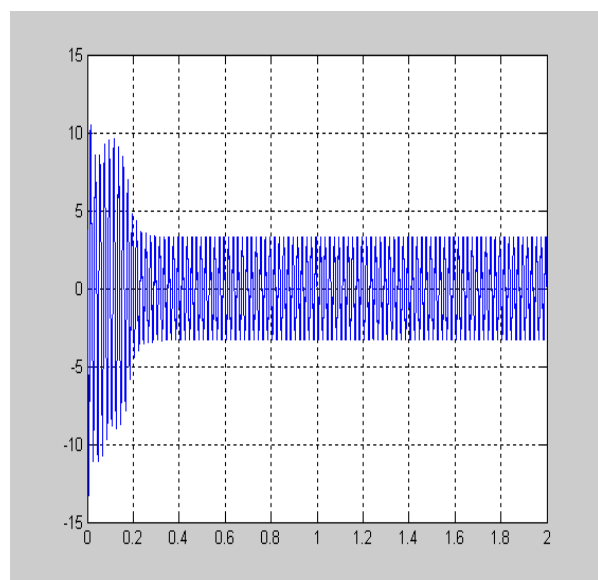
a) *Vitesse (rd/s)*



b) *couple (N.m)*

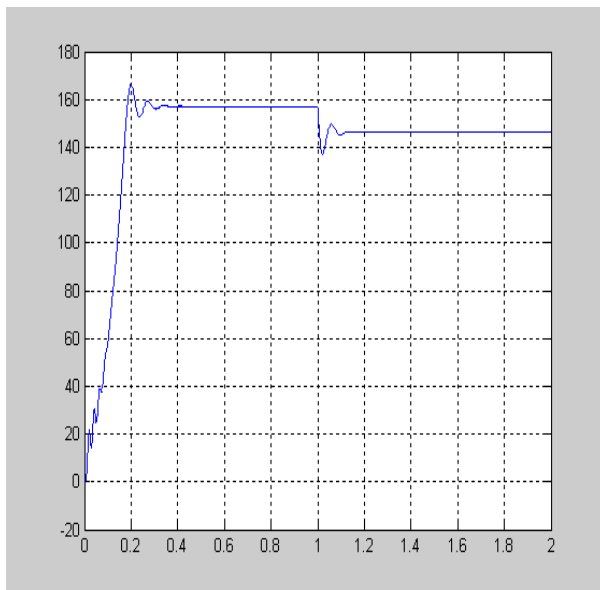
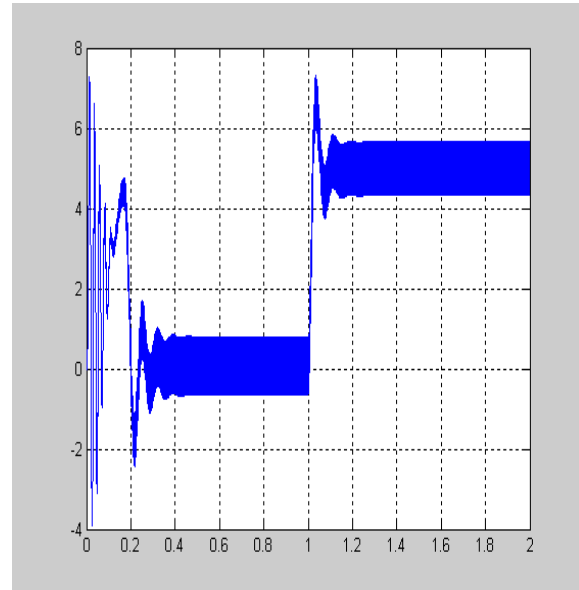
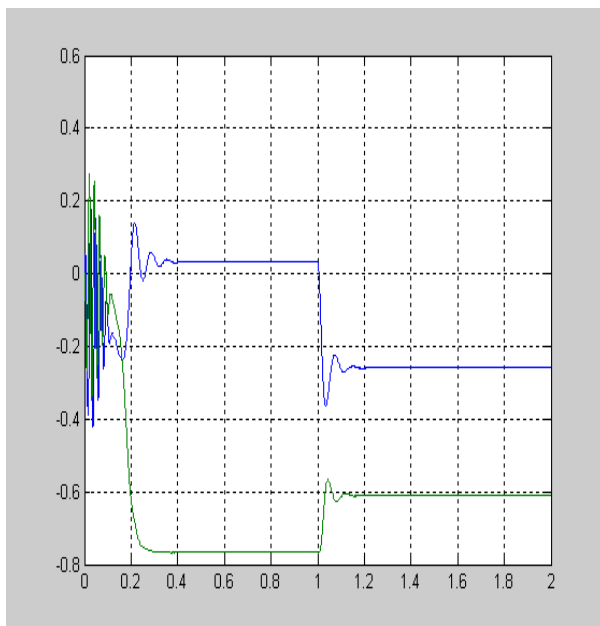


c) *Les flux (wb)*

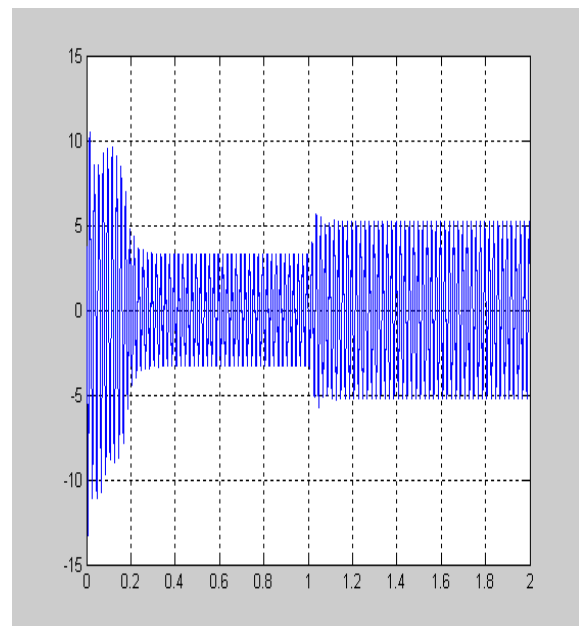


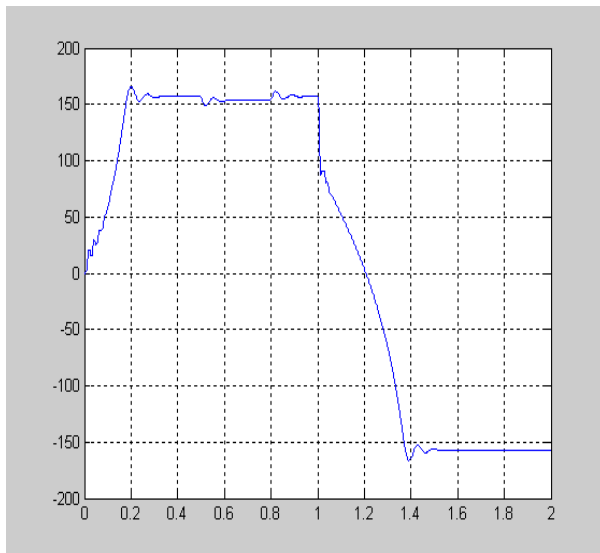
d) *courant ias(A)*

Fig.I.7 Association convertisseur – machine à vide

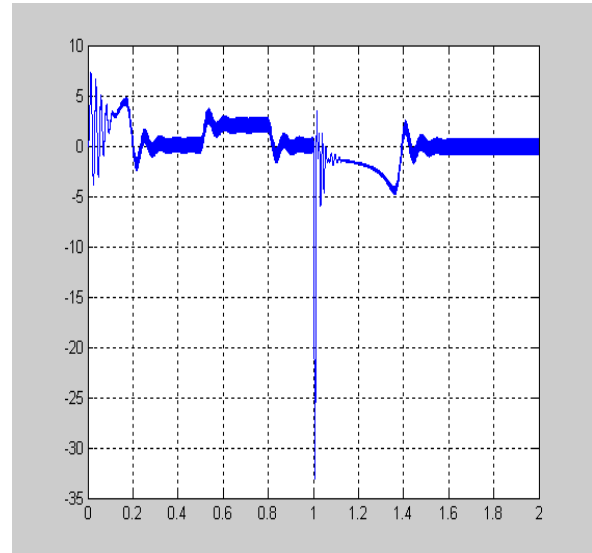
a) Vitesse (rd/s)b) couple (N.m)

c) Les flux

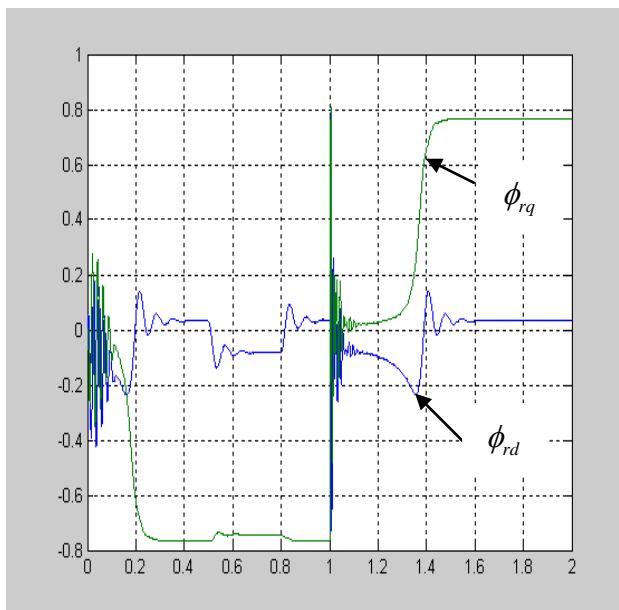
d) courant i_{as} (A)Fig.I.8 Association convertisseur – machine avec une charge de 5 Nm à l'instant $t=1\text{ s}$



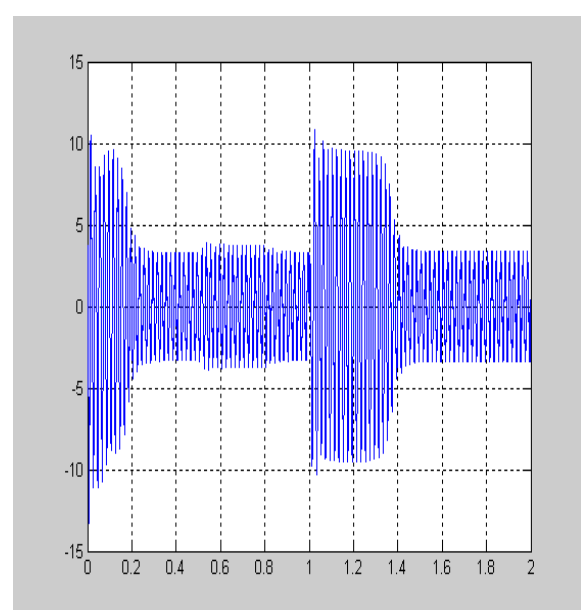
a) Vitesse (rd/s)



b) couple (N.m)



c) Les flux



d) courant ias(A)

Fig.I.9 Association convertisseur – machine avec application d'une charge à $t=0.5s$ et $t=0.8s$ et inversement de sens de rotation.

Chapitre II



Commande vectorielle

COMMANDE VECTORIELLE

II.1 Introduction

L'objectif de ce type de contrôle est d'aboutir à un modèle simple de la machine asynchrone qui rend compte de la commande séparée des grandeurs flux Φ et courant I , générateurs du couple. Il s'agira donc de retrouver la quadrature entre I et Φ , naturellement découplés pour une machine à courant continu (courant d'excitation- producteur de flux, et courant d'induit- producteur de couple). La difficulté va résider justement dans le fait que, pour une machine à induction, il est difficile de distinguer le courant producteur de couple du courant producteur du flux, fortement couplés.

La méthode du flux orienté consiste à choisir un système d'axes (d,q) . qui est un repère tournant biphasé orienté sur ϕ_r (flux rotorique) ou ϕ_s (flux statorique) et un type de commande qui permettent de découpler le couple et le flux [6].

II.2 Principe de la commande vectorielle

Cette méthode est basée sur le choix d'un repère de référence [7], [8]. Lié au flux rotorique. Si on décompose le courant statorique en ses composantes i_{sd} suivant le flux rotorique et i_{sq} en quadrature avec ce flux, on met en évidence une commande découplée du flux et du couple. On obtient alors des fonctionnements comparables à ceux d'une machine à courant continu à excitation séparée ou le courant inducteur contrôle le flux et le courant induit contrôle le couple.

II.3 Choix de référentiel

On peut à priori choisir les axes de référence selon l'un des flux de la machine. A savoir le flux statorique. Le flux rotorique ou le flux d'entrefer et vérifier le degré de découplage entre le flux et le couple selon ce choix

- Soit le flux rotorique avec les conditions : $\phi_{rd} = \phi_r; \phi_{rq} = 0$
- Soit le flux statorique avec les conditions : $\phi_{sd} = \phi_s; \phi_{sq} = 0$
- Soit le flux d'entrefer avec les conditions : $\phi_{md} = \phi_m; \phi_{mq} = 0$

II.3.1 référentiel lié au flux rotorique

L'axe d étant selon le flux rotorique, $\phi_{rd} = \phi_r; \phi_{rq} = 0$

L'expression du couple électromagnétique est Proportionnel au produit du flux par la composante du courant statorique en quadrature avec le flux :

$$C_{em} = p \frac{M_{sr}}{L_r} \phi_{rd} i_{sq} \quad (\text{II.1})$$

. Avec la première équation rotorique, on obtient :

$$\left(1 + T_r \frac{d}{dt}\right) \phi_{rd} = M_{sr} i_{sd} \quad (\text{II.2})$$

Ainsi, le flux ne dépend que de la composante du courant statorique dans l'axe d , d'où un découplage naturel entre le flux et le couple.

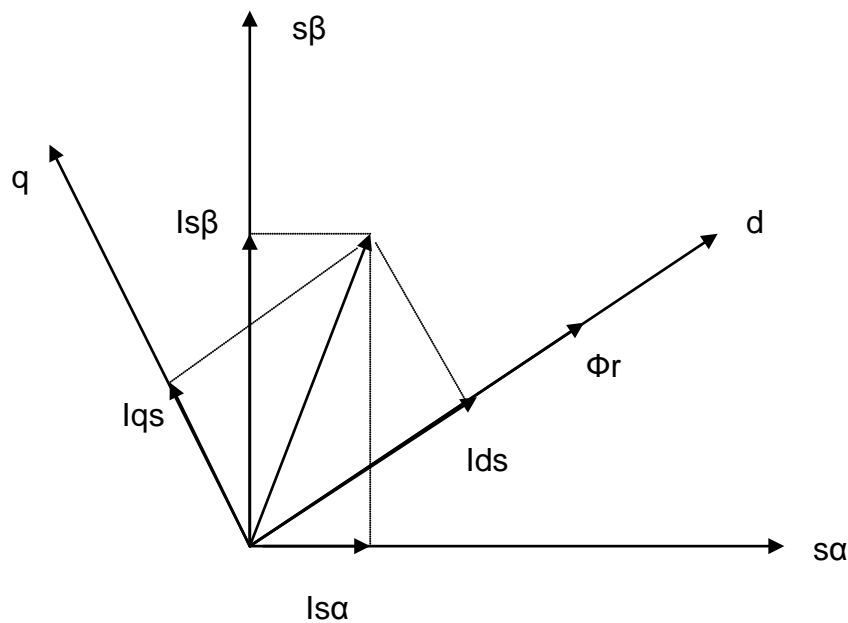


Fig II.1. Repère d q lié au flux rotorique

Dans les trois référentiels, le couple est proportionnel au produit du flux par la composante du courant statorique en quadrature avec le flux. Ainsi dans un fonctionnement à flux constant, cette composante est l'image du couple.

Seul le choix du flux rotorique permet un découplage naturel caractérisé par une indépendance du flux par rapport à la composante du courant statorique en quadrature avec le flux. Le référentiel lié au flux rotorique est choisi pour obtenir des fonctionnements de la machine asynchrone comparables à ceux de la machine à courant continu. Dans la majorité des cas, le référentiel est choisi selon le flux rotorique.

III.4 commande vectorielle

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.

En imposant, $\varphi_{rq} = 0$, les équations de la machine dans un référentiel lié au champs tournant deviennent :

$$V_{ds} = \sigma L_s \frac{di_{ds}}{dt} + \left(R_s + R_r \frac{L_m^2}{L_r^2} \right) i_{ds} - \omega_s \sigma L_s i_{qs} - \frac{L_m}{L_r} R_r \phi_r \quad (\text{II.3})$$

$$v_{qs} = \sigma L_s \frac{di_{qs}}{dt} + \left(R_s + R_r \frac{L_m^2}{L_r^2} \right) i_{qs} + \omega_s \sigma L_s i_{ds} - \frac{L_m}{L_r} \omega_m \phi_r \quad (\text{II.4})$$

$$\tau_r \frac{d\phi_r}{dt} + \phi_r = M I_{ds} \quad (\text{II.5})$$

$$\omega_r = \frac{M}{\tau_r \phi_r} I_{qs} \quad (\text{II.6})$$

$$C_e = p \frac{M}{L_r} \phi_r I_{qs} \quad (\text{II.7})$$

on distingue la commande vectorielle directe et indirecte [6]. Dans la *commande indirecte*, 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 ω_r . En ce qui concerne la *commande directe*, l'angle de Park est calculé directement à l'aide des grandeurs mesurées ou estimées.

III.5 contrôles vectoriels direct et indirect

Dans le contrôle vectoriel direct [7] [9] [10] [11], on effectue une régulation du flux qui nécessite la connaissance de celui-ci, alors que dans le contrôle vectoriel indirect, on s'affranchit de la connaissance de ce flux en faisant quelques approximations.

III.5.1 contrôle vectoriel indirect

Dans cette méthode, on ne régule pas le flux rotorique et on n'a donc besoin ni de capteur, ni d'estimateur ou d'observateur de flux. Si l'amplitude du flux rotorique réel n'est pas utilisée, sa position doit être connue pour effectuer les changements de coordonnées. Ceci exige la présence d'un capteur de position du rotor. Considérons le flux rotorique ϕ_r et le couple C_{em} comme référence de commande.

La structure de commande du schéma bloc est la suivante :

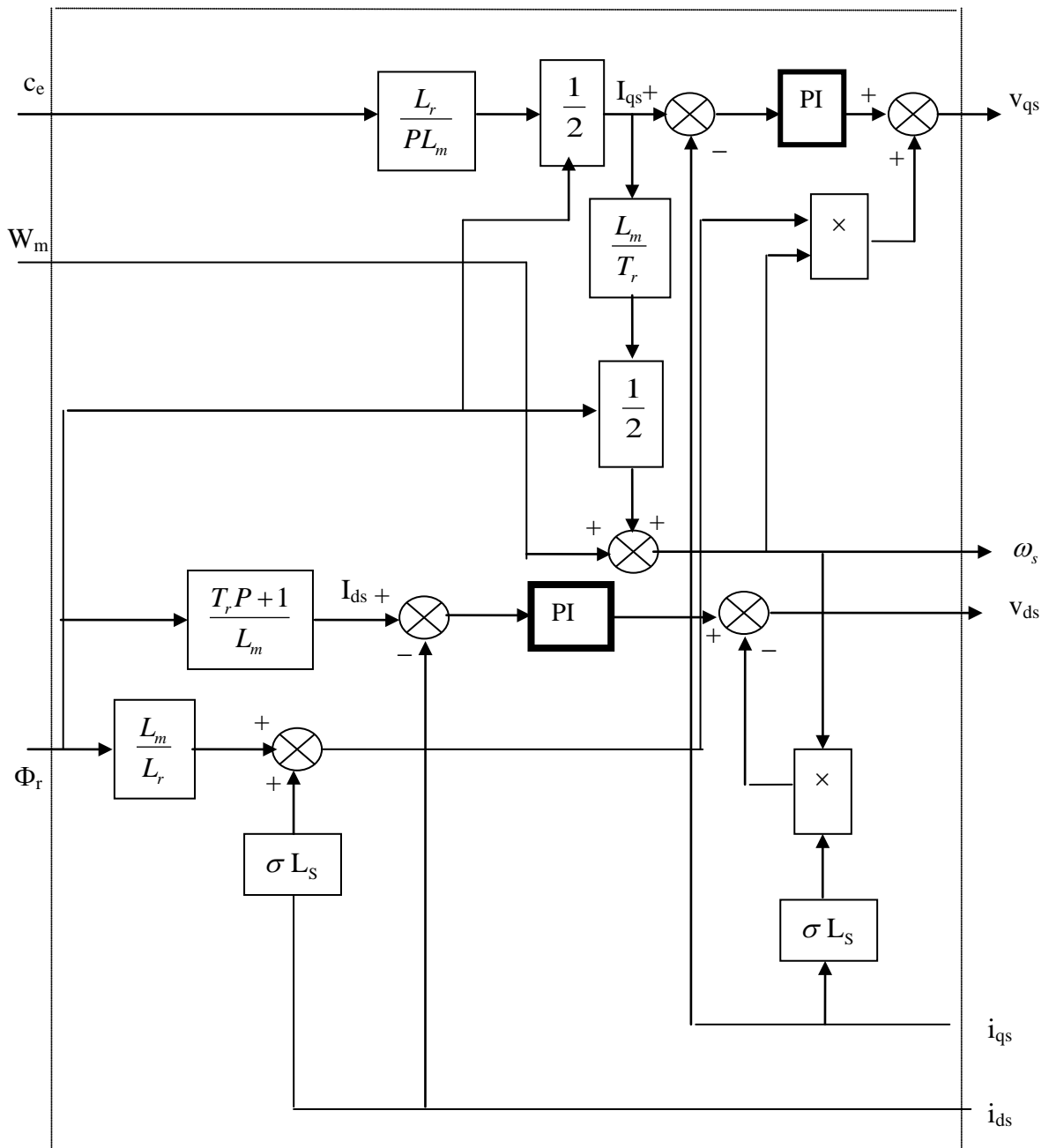


Fig. II.2 Structure du bloc F.O.C

II.5.2 Bloc de défluxage

Pour obtenir un variateur de vitesse, il suffit d'adjoindre à la boucle ouverte un régulateur de flux et un régulateur de couple. La vitesse est mesurée et il est donc possible de faire sa régulation. Par contre le flux ne peut pas être mesuré directement et son contrôle, le plus simple est du type réaction le bloc de défluxage est défini par la non linéarité suivante :

$$\Phi_r = \begin{cases} \Phi_m & \text{si } |\Omega| \leq \Omega_n \\ \Phi_m \left(\frac{\Omega_n}{\Omega} \right) & \text{si } |\Omega| \geq \Omega_n \end{cases}$$

(II.10)

il assure le fonctionnement à couple constant pour les vitesses inférieures à la vitesse nominale et au-delà de cette valeur il permet de diminuer le couple. Dans ce cas, il assure un fonctionnement à tension ou à la puissance constante.

II.6 Régulation méthodes classiques

II.6.1 Introduction

La commande vectorielle utilisée est une commande indirecte par orientation du flux rotorique. Il est nécessaire d'ajouter des termes de compensation afin de rendre les axes d et q complètement indépendants. Ce découplage permet surtout d'écrire les équations de la machine et de la partie régulation d'une manière simple et ainsi de calculer les coefficients des régulateurs.

II.6.2 Découplage

Définissons deux nouvelles variables de commande v_{sd1} et v_{sq1} [12], telle que :

$$v_{sd} = v_{sd1} - e_{sd} \quad \text{Et} \quad v_{sq} = v_{sq1} - e_{sq}$$

$$e_{sd} = \omega_s \sigma L_s i_{qs} + \frac{L_m}{L_r} R_r \phi_r \quad (\text{II.11})$$

$$e_{qs} = -\omega_s \sigma L_s i_{ds} - \frac{L_m}{L_r} \omega_s \phi_r + \frac{L_m}{L_r T_r} i_{qs} \quad (\text{II.12})$$

Les tensions v_{ds} et v_{qs} sont alors reconstituées à partir des tensions v_{ds1} et v_{qs1} (figure

II.3)

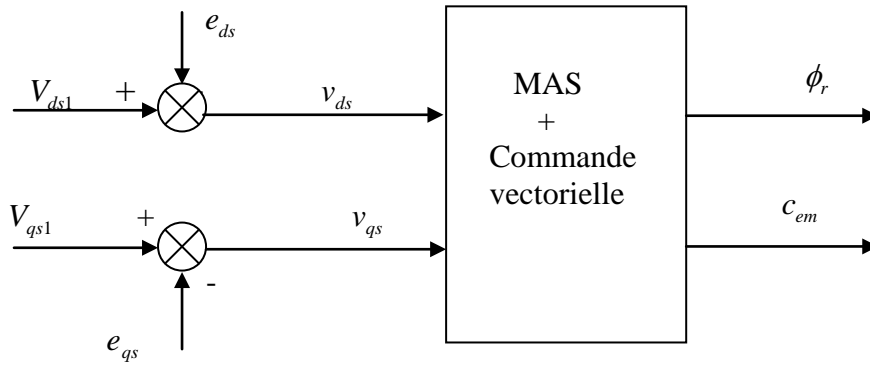


Figure II.3 reconstitution des tensions v_{ds} et v_{qs}

En faisant apparaître de manière explicite le flux (II.8) et le couple (II.9), nous obtenons :

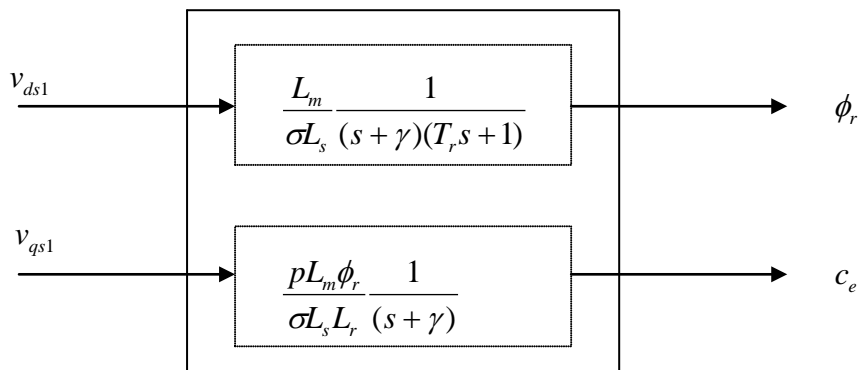


Figure II.4 commande découplée – expression de ϕ_r et c_{em}

II.6.3 Schéma fonctionnel

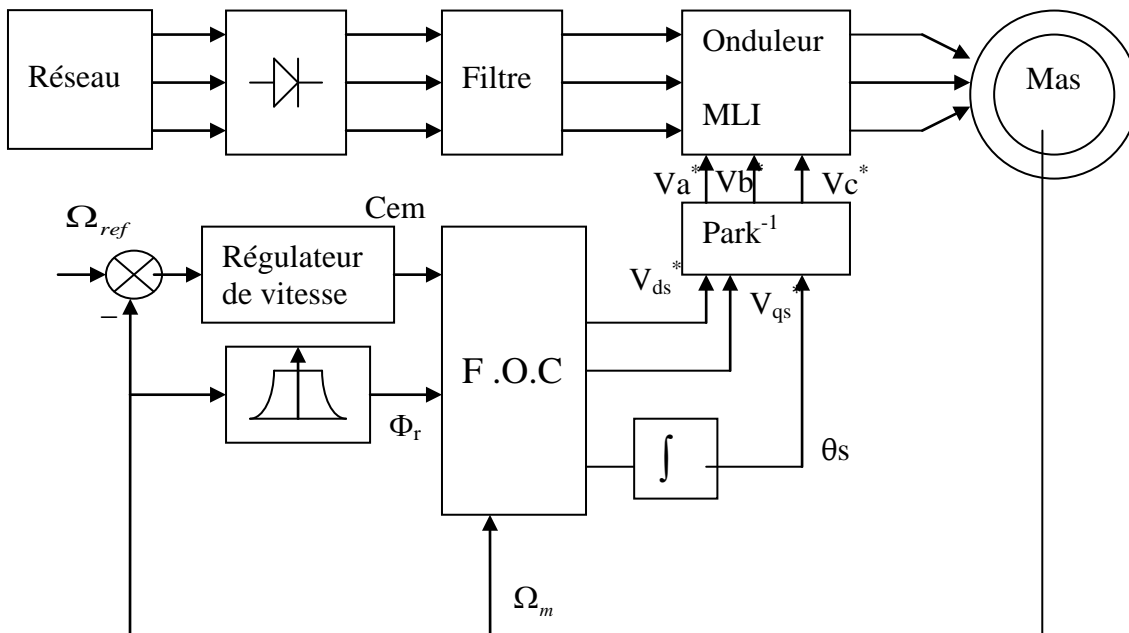


Fig. II.5 Schéma bloc de la commande vectorielle indirecte.

La structure de commande est constituée de l'association d'une machine asynchrone avec onduleur de tension alimentée par un pont à diodes à travers un filtre passe-bas; les tensions de sortie de l'onduleur sont contrôlées par la technique de modulation de Largeur d'Impulsion (MLI) qui permet le réglage simultané de la fréquence et de tension de sortie de l'onduleur. La stratégie triangulo-sinusoidale, retenue le long de notre travail, est l'une des techniques les plus simples. Elle consiste à comparer les trois tensions de référence (V_a^*, V_b^*, V_c^*), calculées par une transformation inverse de PARK à partir de ($V_{ds}^*, V_{qs}^*, \theta_s$) un signal triangulaire d'amplitude fixe et de pulsation nettement supérieur à ω_s .

II.7 Ajustement des régulateurs

II.7.1 Régulateur boucle interne du courant I_{ds}

Le régulateur de courant direct permet de définir la tension V_{ds}^* , nécessaire pour maintenir le flux de référence à flux constant égale à Φ_r^* , en supposant que le découplage est réalisé, on aura:

$$\frac{i_{ds}^r(s)}{V_{ds}^r(s)} = \frac{1}{R_s + \sigma L_s S} \quad (\text{II.13})$$

La boucle de régulation du courant i_{ds} peut se représenter par le schéma bloc suivant:

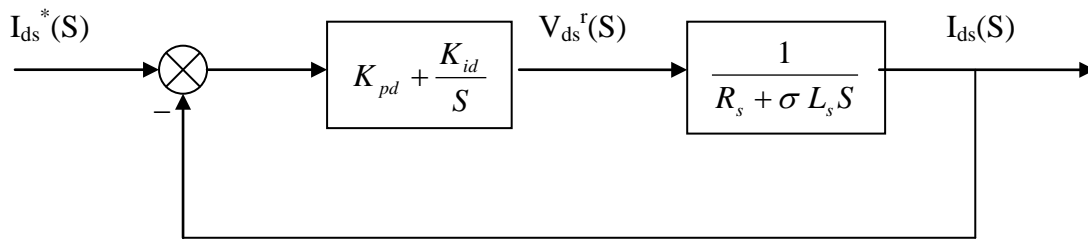


Fig. II.6 schéma fonctionnel de régulation de courant i_{ds} .

L'expression du courant de référence est déterminée à partir de la boucle de réglage de flux. La fonction de transfert en boucle fermée est la suivante:

$$\frac{i_{ds}^r(s)}{i_{ds}^*(s)} = \frac{(k_{pd}S + K_{id}) \frac{1}{\sigma L_s}}{S^2 + \frac{R_s + K_{pd}}{\sigma L_s} S + \frac{k_{id}}{\sigma L_s}} \quad (\text{II.14})$$

$$P(s) = S^2 + \frac{R_s + K_{pd}}{\sigma L_s} S + \frac{K_{id}}{\sigma L_s} = 0 \quad (\text{II.15})$$

En posant au polynôme caractéristique deux pôles afin d'avoir en boucle fermée

- Un amortissement $\xi = \frac{1}{2}$, (critère d'ajustement optimale)
- Une pulsation $\omega_0 = \frac{6\xi}{T_{rep}}$

D'où l'équation caractéristique en boucle fermée est de la forme suivante:

$$Q(S) = S^2 + 2\xi \omega_0 S + \omega_0^2 = 0 \quad (\text{II.16})$$

L'équation terme à terme des deux équations (II.15) et (II.16) entraîne que

$$K_{pd} = 2\sigma L_s \xi \omega_0 - R_s \quad \text{et} \quad K_{id} = \sigma L_s \omega_0^2 \quad (\text{II.17})$$

II.7.2 Réglage de la boucle interne du courant i_{qs}

Habituellement, les schémas de régulation de la vitesse sont constitués de deux étages, le premier comprend les boucles de régulation des courants, et le second; externe au premier, est destiné à la régulation de la vitesse. Ceci est justifié par le fait qu'en général on considère que la dynamique de la vitesse est très lente devant la dynamique des courants, c'est pour cette raison que, dans les régulations classiques, la vitesse est toujours supposé constante pendant que les courants sont réglés.

Les mêmes calculs sont effectués pour dimensionner le régulateur du courant i_{sq} . Si on impose la même dynamique en boucle fermée. Les coefficients k_{iq} et K_{pq} seront identiques à ceux du régulateur du courant i_{sd} .

II.7.3 Boucle fermée de régulation de la vitesse

Le régulateur de vitesse permet de déterminer le couple de référence; afin de maintenir la vitesse correspondante pour que la cascade soit justifiée; il faut que la boucle interne soit l'équation mécanique donnée.

$$\frac{w_m(s)}{w_m^*(s)} = \frac{P}{k_f + JS} \quad (\text{II.18})$$

En associant à cette fonction un régulateur PI, il arrive:

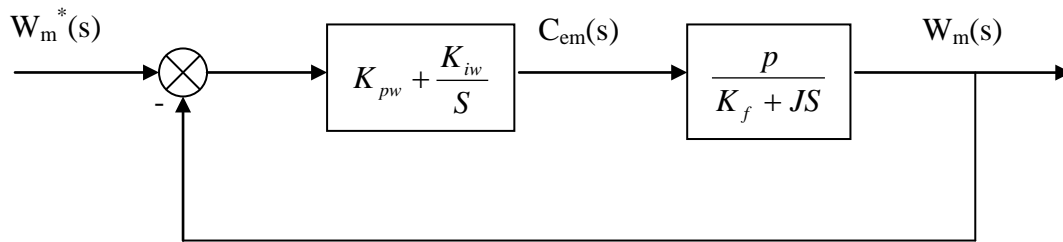


Fig. II.7 schéma fonctionnel de régulation de vitesse.

La fonction de transfert en boucle fermée, calculée à partir du schéma précédent, est donnée par l'expression suivante:

$$\frac{w_m(s)}{w_m^*(s)} = \frac{(K_{pw}S + K_{iw})P}{Q(s)} \quad (\text{II.19})$$

L'équation caractéristique $Q(s)$ est:

$$Q(s) = s^2 + \left(\frac{K_f S + K_{pw}}{J} \right) s + \frac{K_{iw}P}{J} \quad (\text{II.20})$$

En imposant toujours des pôles correspondants aux critères utilisés, en boucle fermée et, par identification avec la nouvelle équation caractéristique désirée, on arrive sans difficulté à:

$$K_{iw} = \frac{Jw_0^2}{P}, \quad k_{pw} = \frac{(2\xi w_0 J - K_f)}{P} \quad (\text{II.21})$$

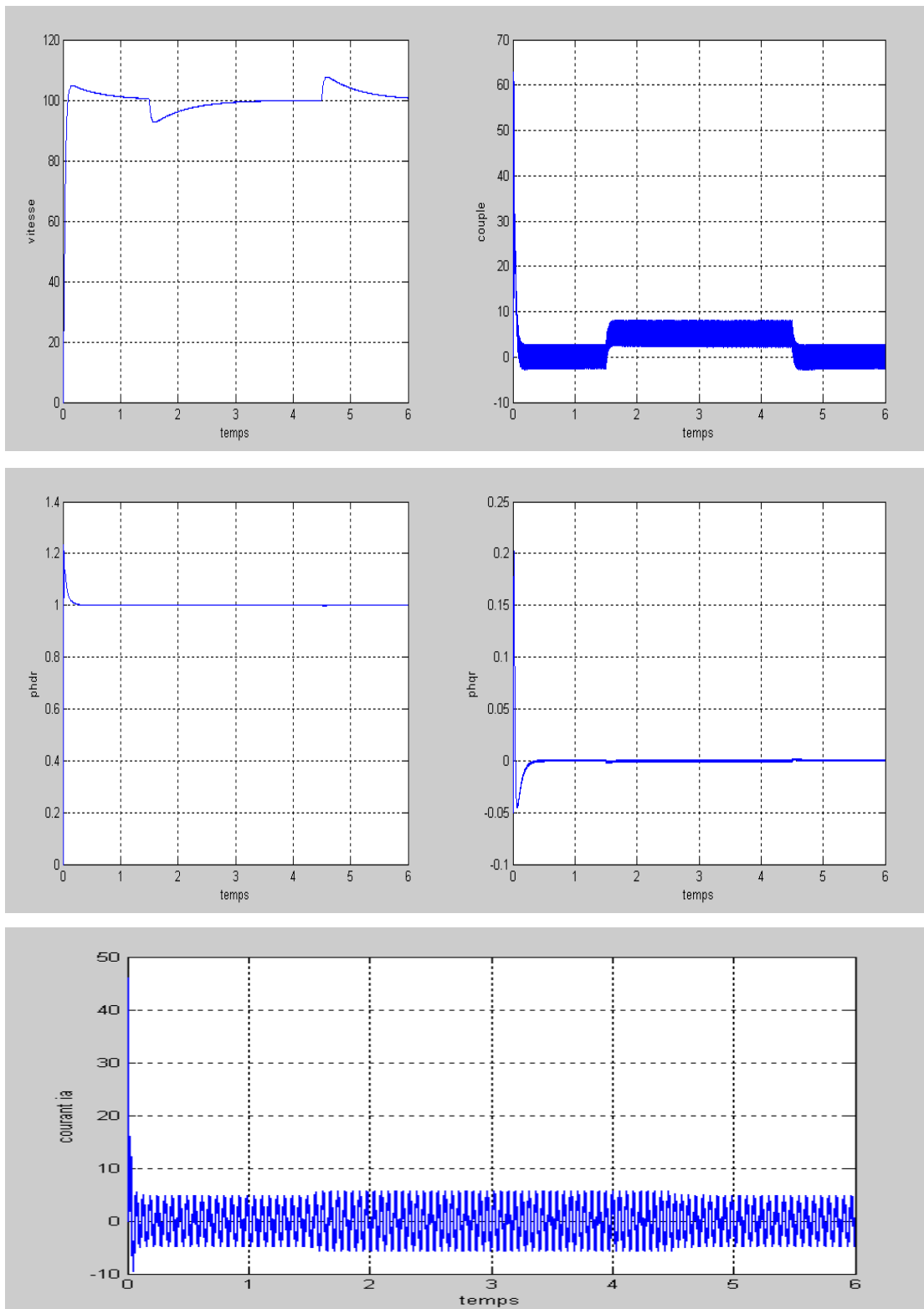


Fig. II.8 commande vectoriel indirect, application d'une charge à $t=1.5s$ et $t=4.5s$ avec une vitesse de 100 rd/s

III.8 contrôle vectoriel direct

Cette méthode nécessite une bonne connaissance du module du flux et de sa phase et celle-ci doit être vérifiée quelque soit le régime transitoire effectué. Il faut donc procéder à une série de mesures aux bornes du système. Une première possibilité est de mettre des capteurs de flux dans l'entrefer et de mesurer directement les composantes ϕ_{dm} et ϕ_{qm} de manière à en déduire l'amplitude de la phase. Les capteurs, mécaniquement fragile, sont soumis à des conditions sévères dues aux vibrations et aux échauffements. Les signaux captés sont entachés des harmoniques d'encoche et leur fréquence varie avec la vitesse ce qui nécessite des filtres ajustables.

La précision de la définition du flux dépend des paramètres inductifs, affectés par la saturation du circuit magnétique. La phase des courants est déterminée à partir de l'angle de charge défini par le courant i_{qs} et la position angulaire du flux rotorique. La mesure directe permet de connaître exactement la position du flux. Ce mode de contrôle garantit un découplage correct entre le flux et le couple quel que soit le point de fonctionnement. Toutefois, il nécessite l'utilisation d'un moteur équipé de capteurs de flux, ce qui augmente considérablement le coût de sa fabrication et rend plus fragile son utilisation. Dans la grande majorité des cas, on ne dispose pas de capteurs de flux et on fait donc appel à des estimateurs ou à des observateurs.

III.8.1 Estimateur

L'estimateur est obtenu à partir des équations suivantes :

$$\phi_r^e + T_r \frac{d\phi_r^e}{dt} = L_m i_{ds}$$

$$c_{em}^e = p \frac{L_m}{L_r} i_{qs} \phi_r^e \quad (\text{II.22})$$

$$\omega_{gl}^e = \omega_s - \omega_m = \frac{L_m}{T_r \phi_r^e} i_{qs}$$

$$\theta_s = \int \omega_s dt$$

III.8.1.1 Calcul de ϕ_r

Les grandeurs d'état ou de sortie utilisées pour l'élaboration de la commande sont souvent difficilement accessibles pour des raisons techniques (c'est le problème du flux), le flux peut être reconstitué par :

- des estimateurs utilisés en boucle ouverte,
- des observateurs corrigeant en boucle fermée les variables estimées.

Les estimateurs reposent sur l'utilisation d'une représentation de la machine sous forme d'équations de Park définie en régime permanent (estimateur statique) ou transitoire (estimateur dynamique). Ils sont obtenus par une résolution directe des équations associées à ce modèle.

Le système d'équations (II.23) permet d'estimer le flux ϕ_r

$$\phi_r^e = \frac{Lm}{1+Tr} i_{ds} \quad (\text{II.23})$$

III.8.1.2 Calcul de ω_s et θ_s

La pulsation statorique s'écrit :

$$\omega_s = p\Omega + \frac{Lm}{Tr} \frac{i_{qs}}{\phi_r} \quad (\text{II.24})$$

L'équation n'est pas exploitable telle quelle, puisque ϕ_r est nul au démarrage du moteur.

Nous utiliserons, pour l'implantation, l'équation suivante :

$$\omega_s = p\Omega + \frac{Lm}{Tr} \frac{i_{qs}}{\phi_r + \varepsilon} \quad (\text{II.25})$$

Avec $\varepsilon = 0.01$

Nous avons alors :

$$\theta_s = \frac{1}{s} \omega_s \quad (\text{II.26})$$

III.8.2 Schéma complet de la commande vectorielle directe à flux rotorique orienté

Le schéma que nous proposons (figure II.8) est une commande vectorielle de type direct : le flux rotorique est asservi à une consigne de flux. Une commande indirecte ne comporterait pas de régulateur de flux.

Nous utilisons les estimateurs de flux et de pulsation statorique déterminés précédemment

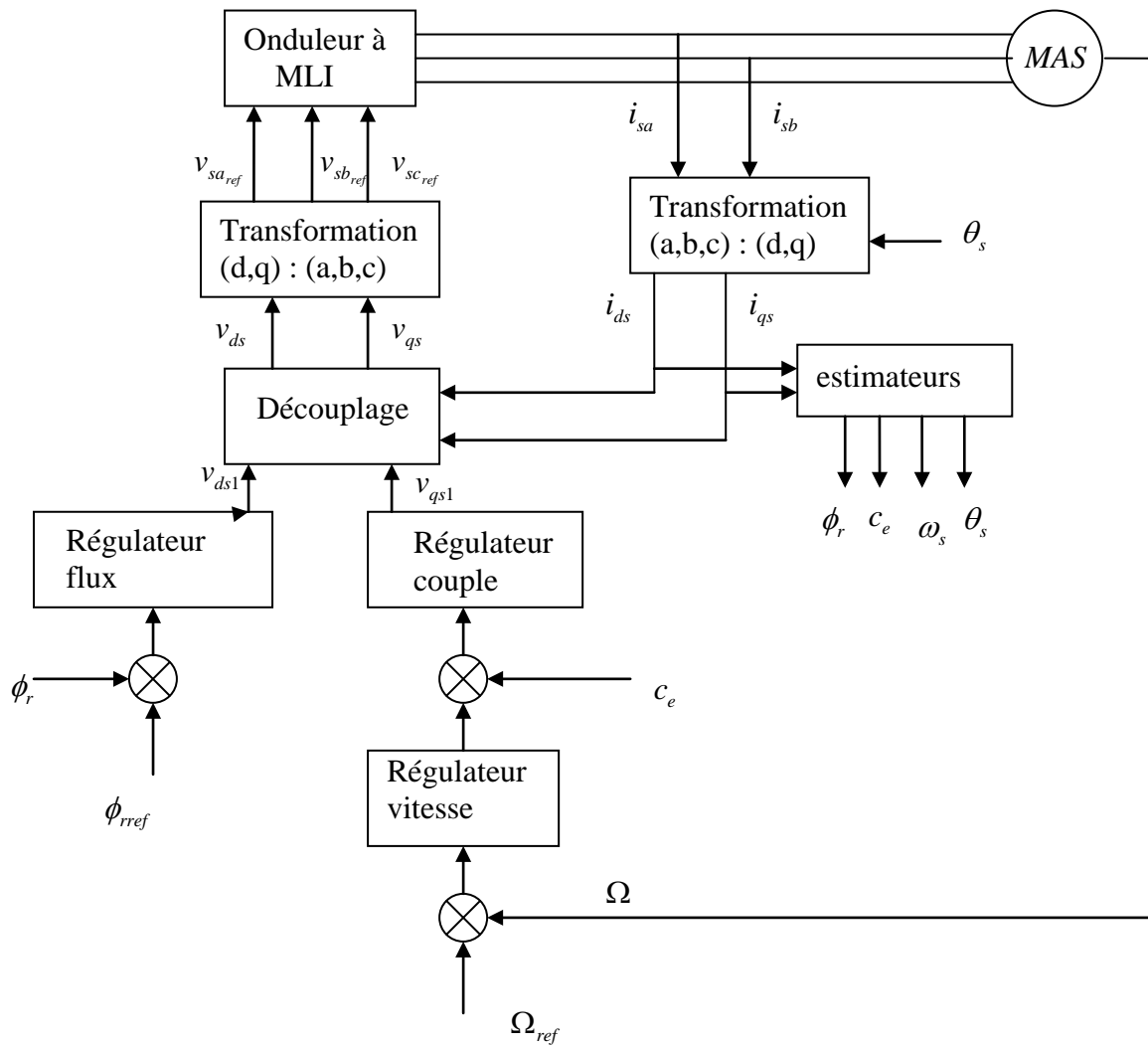


Figure II.9 commande vectorielle directe de flux d'une machine alimentée en tension

III.8.3 Calcul des régulateurs

III.8.3.1 Régulateur de flux

Le découplage proposé (figure II.1) permet d'écrire :

$$\phi_r = \frac{k_1}{(s + \gamma)(s + \frac{1}{T_r})} v_{ds1} \quad (\text{II.27})$$

Avec $k_1 = \frac{L_m}{\sigma L_s T_r}$ et $\gamma = \frac{R_s + R_r \frac{L_m^2}{L_r^2}}{\sigma L_s}$

Soit un régulateur proportionnel intégral de type :

$$w(s) = k_p + \frac{k_i}{s}$$

Nous pouvons représenter le système en boucle ouverte par la figure (II.9)

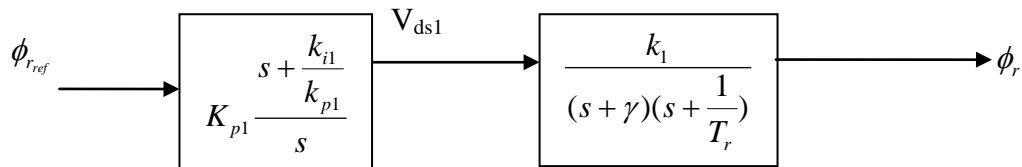


Figure II.10 schéma en boucle ouverte

Compensons le pôle le plus lent par le numérateur de la fonction de transfert de notre

régulateur, soit $(s + \frac{1}{T_r})$ par $(s + \frac{k_{i1}}{k_{p1}})$, ce qui traduit par la condition :

$$\frac{k_{i1}}{k_{p1}} = \frac{1}{T_r} \quad (\text{II.28})$$

En boucle ouverte, la fonction de transfert s'écrit maintenant :

$$w_{BO}(S) = \frac{k_{p1} k_1}{s(s + \gamma)}$$

L'équation caractéristique du système en boucle fermée est la suivante :

$$1 + \frac{\gamma}{k_{p1} k_1} s + \frac{s^2}{k_{p1} k_1}$$

Que nous cherchons à identifier à la forme canonique du 2^{ème} ordre.

Nous avons donc à résoudre le système suivant :

$$\left\{ \begin{array}{l} \frac{1}{k_{p1}k_1} = \frac{1}{\omega_n^2} \\ \frac{\gamma}{k_{p1}k_1} = \frac{2\zeta_1}{\omega_n} \end{array} \right. \quad \text{et} \quad \left\{ \begin{array}{l} \frac{\gamma}{k_{p1}k_1} = \frac{2\zeta_1}{\omega_n} \end{array} \right.$$

Le gain k_{p1} est donné par :

$$k_{p1} = \frac{\gamma^2}{k_1(2\zeta_1)^2} \quad (\text{II.29})$$

L'équation (II.28) permet de déduire k_{i1}

$$k_{i1} = \frac{1}{T_r} k_{p1} \quad (\text{II.30})$$

III.8.3.2 Régulateur de couple

De même, les équations proposées figure (II.4) permettent d'exprimer c_e :

$$c_e = \frac{k_2}{s + \gamma} v_{qs1} \quad \text{Avec} \quad k_2 = \frac{pL_m \phi_{rref}}{\sigma L_s L_r} \quad (\text{II.31})$$

Les équations du régulateur seront dépendantes de la consigne de flux ϕ_{rref}

L'utilisation d'un régulateur Proportionnel- Intégral donne le schéma en boucle ouverte suivante

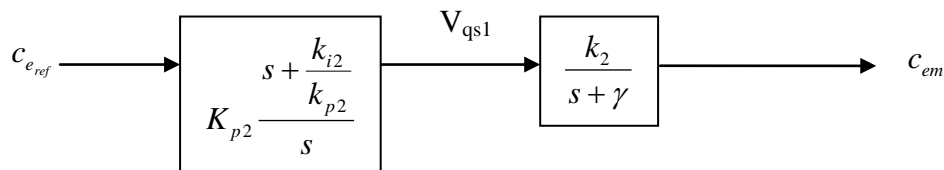


Figure II.11 schéma en boucle ouverte

Compensons le pôle $(s + \gamma)$ par $(s + \frac{k_{i2}}{k_{p2}})$, ce qui traduit par la condition :

$$\frac{k_{i2}}{k_{p2}} = \gamma \quad (\text{II.32})$$

En boucle ouverte, la fonction de transfert s'écrit maintenant :

$$w_{BO}(s) = \frac{k_{p2}k_2}{s}$$

En boucle fermée, nous obtenons une réponse de type 1^{er} ordre de constant de temps $\frac{1}{k_{p2}k_2}$

Pour un temps de réponse imposé t_{rep} , nous obtenons la condition suivante :

$$3 \frac{1}{k_{p2}k_2} = t_{rep}$$

Soit
$$k_{p2} = 3 \frac{\sigma L_s L_r}{p L_m \phi_{r_{ref}} t_{rep}} \tag{II.33}$$

Et, d'après l'équation (II.33) :

$$k_{i2} = \gamma k_{p2} \tag{II.34}$$

III.8.3.3 Régulateur de vitesse

La chaîne de régulation de vitesse peut être représentée par le schéma fonctionnel suivante :

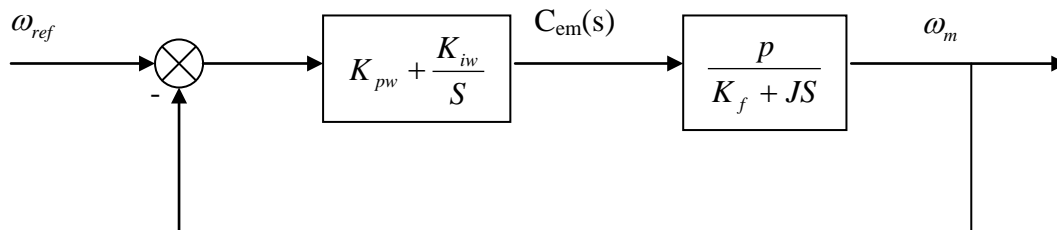


Figure II.12 schéma en boucle fermée

L'utilisation d'un régulateur PI classique donne :

$$\omega_m = \frac{1}{Js + f} \left(\frac{k_{p3}s + k_{i3}}{s} \right) (\omega_{ref} - \omega_m)$$

Cette fonction de transfert possède une dynamique du 2nd ordre.

En identifiant le dénominateur à la forme canonique $\frac{1}{1 + \frac{2\zeta}{\omega_n}s + \frac{s^2}{\omega_n^2}}$, nous avons à résoudre

système d'équation suivant :

$$\frac{J}{k_{i3}} = \frac{1}{\omega_n^2} \tag{II.35}$$

$$\frac{2\zeta_3}{\omega_n} = \frac{k_{p3}}{k_{i3}} \tag{II.36}$$

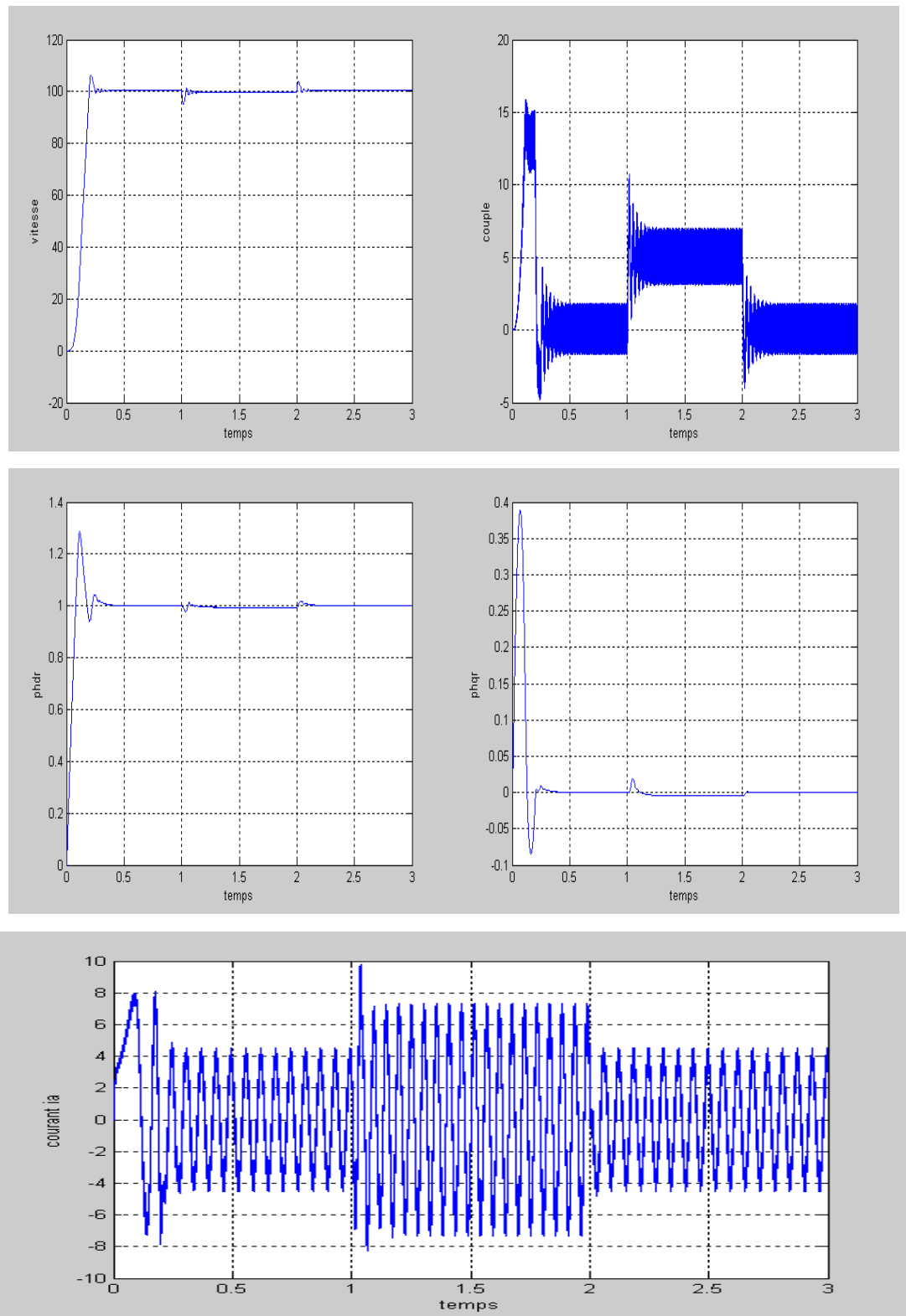


Figure II.13 commande vectoriel direct, application d'une charge à $t=1s$ et $t=2s$ avec une vitesse de 100 rd/s

III.9 Simulation et interprétation

La figure (II.8) montre l'évolution de la vitesse de rotation et les flux rotorique sur les deux axes d et q, on note qu'il y a une découplage entre les flux et le couple malgré l'application de la charge, c'est la commande vectorielle indirecte par orientation du flux rotorique.

La figure (II.13) c'est la commande directe du flux, on note que les deux flux et couple sont presque découplés.

III.10 Conclusion

L'application de la technique de la commande vectorielle permet d'assimiler la machine asynchrone à une machine à courant continu qui est à commander. Le flux et le couple sont commandés alors séparément. L'allure du flux rotorique selon l'axe (d) suit l'allure du courant I_{ds} , le flux selon l'axe (q) est nul et l'allure du couple électromagnétique suit l'allure du courant I_{qs} , La méthode indirecte de la commande vectorielle est simple, car elle permet de contrôler la position du flux par l'intermédiaire d'un bloc appelé F.O.C. La méthode directe nécessite la connaissance de la valeur du flux rotorique ce qui est toujours difficile à obtenir et conduit aussi à utiliser un capteur de flux.

Chapitre III



Commande par logique floue

COMMANDE PAR LOGIQUE FLOUE

III.1 principe de la logique floue

Présenté par le philosophe MAX BLACK dès 1937, le concept de la logique floue fut réellement introduit en 1965 par Lotfi Zadeh, un professeur d'électronique à l'université de Berkeley (USA), sa théorie `` *fuzzy set theory* `` n'eut pas un succès immédiat, elle fut développée surtout en Europe et au Japon. L'idée de base a pris naissance lorsqu'on a constaté la difficulté de programmer un automate en vue de sa réalisation d'une tâche, cependant jugée simple à réaliser par un être humain.

Le mode de raisonnement humain et le moyen de formaliser la connaissance humaine dans un langage accessible à une machine constituent, donc, les deux principaux sujets de réflexion qui ont conduit à l'apparition de la logique floue, freinée par la réticence à appréhender un nouveau concept et par des difficultés de réalisation pratique, la logique floue ne débouche réellement qu'en 1990 avec l'apparition de produits et de systèmes utilisant cette technique [13]-[14]. Aujourd'hui, la logique floue est arrivée à maturité et est utilisée dans de nombreux produits destinés au `` *grand public* `` notamment au Japon. Sa mise en œuvre est maintenant facilitée par la disponibilité de microprocesseurs dédiés et d'outils puissants de développement.

Le concept de la logique floue vient de la constatation que la variable booléenne, qui ne peut prendre que deux valeurs (vrais ou faux) est mal adaptée à la représentation de la plupart des phénomènes courants. En effet, cette logique floue classique considère qu'une proposition est soit vraie soit fautive. Cela permet de répondre à de nombreuses situations. En revanche, la logique floue distingue une infinité de valeurs de vérité (entre 0 et 1) [20]. Prenons l'exemple d'un système de chauffage. S'il fait froid, il chauffe, sinon il ne chauffe pas. Ensuite, il faut définir 'froid' par exemple, en dessous de 18°C. Donc, le système de chauffage va chauffer à pleine puissance jusqu'à 18°C puis s'arrêter car 'froid' n'est plus 'vrai'. Quelque temps plus tard, la température étant descendue en dessous de 18°C, il va repartir à pleine puissance pour quelque minute. Un tel cycle peut être évité grâce à la logique floue en définissant 'très froid' en dessous de 10°C, le chauffage chauffe alors à pleine puissance et 'non froid' au de la de 20°C, température à laquelle le chauffage s'arrête. Entre 10°C et 20°, le chauffage va recevoir une instruction de chauffage modérée adaptée à la température, le chauffage va alors se stabiliser à une température constante apportant plus de confort qu'une alternance de cycle de chauffage. Lorsque plusieurs paramètres doivent être considérés (par exemple le taux d'humidité ou l'heure

de la journée), la logique floue apporte aussi une réponse plus adaptée car un chauffage traditionnel programmé pour démarrer en dessous de 15°C ou à 18h ne va pas démarrer à 17h59 s'il fait 15.1°C et vous aurez en rentrant alors qu'un chauffage fonctionnant sur le principe de la logique floue comprendra qu'un démarrage modéré est nécessaire.

III.2 Domaine d'application

Les principaux domaines de recherche et d'application de la logique floue sont les suivants :

- Automatisation de la production de fer et de l'acier, purification de l'eau, chaîne de montage et robots de fabrication ;
- Commande des instruments (capteur, et instruments de mesure), et reconnaissance de voix et de caractères ;
- Conception, jugement et décision (consultation, investissement et développement horaires de train) ;
- Commande des unités arithmétiques, des micro- ordinateurs, et réalisation des opérateurs ;
- Traitement d'information telles que les données, recherche de l'information modélisation des systèmes,... etc.

III.3 Structure générale d'un système flou

Chaque système sur la logique floue est composé de quatre blocs principaux [15], [16], [13], [14] :

- Un bloc de connaissance sous forme de règles floues et paramètres des fonctions d'appartenance ;
- Un bloc de décision ou moteur d'inférence ;
- Une fuzzification qui consiste en la transformation des entrées précises en degrés d'appartenance ;
- Une défuzzification qui consiste en la transformation des résultats flous en des sorties précises.

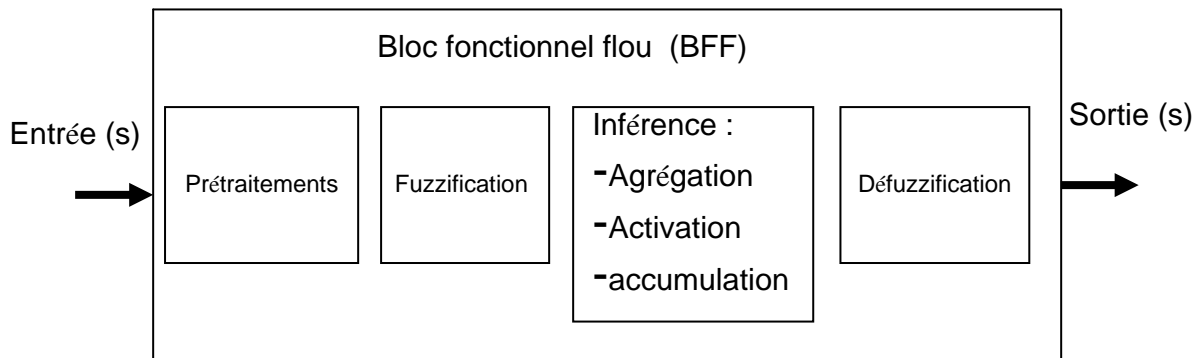


Fig. III.1 schéma générale d'un bloc fonctionnel flou.

Cette définition des blocs fonctionnels flous les rend compatibles avec les langages de programmation tels que le Grafcet est les diagrammes fonctionnels,... etc.

Les BFFs peuvent alors être utilisés comme des sous programmes par ces langages et peuvent être aussi intégrés dans tous les automatismes.

III.3.1. Eléments de base de la logique floue

A l'inverse de la logique booléenne, la logique floue permet à une condition, d'être en un autre état que vrai au faux. Il y a plusieurs degrés de vérification d'une condition. Considérons par exemple la vitesse d'un véhicule sur une route nationale. La vitesse normale est de 90 km/h. une vitesse peut être considérée comme élevée au dessus de 100 Km/h, et comme pas du tout élevée en dessous de 80 Km/h. la logique booléenne envisagerait. Les choses de la manière suivante [20] :

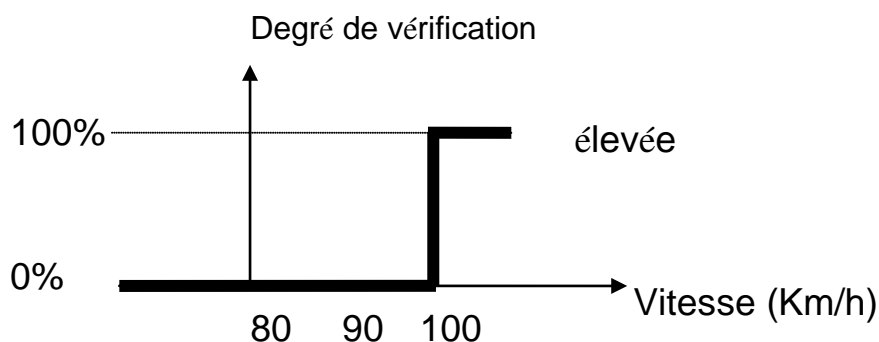


Fig. III.2 Niveaux d'appartenance en logique booléenne.

La vitesse est considérée à 100% comme élevée à partir de 100 Km/h, et 0% en dessous. La logique floue, à l'inverse, permet des degrés de vérification de la condition " la vitesse est – elle élevée"

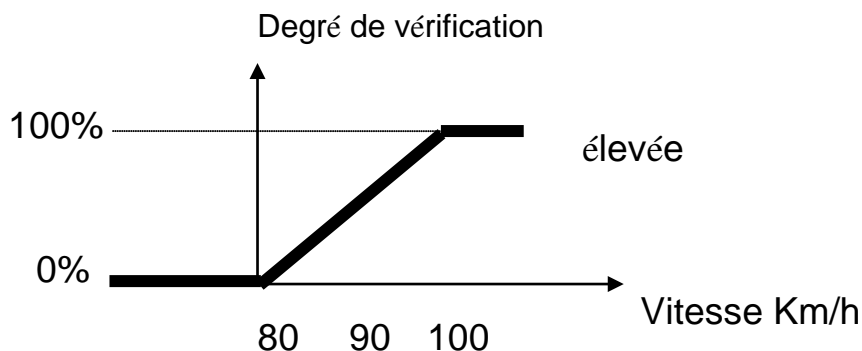


Fig. III.3 Niveaux d'appartenance en logique floue

La vitesse est considérée comme pas du tout élevée en dessous de 80 Km/h. on peut donc dire qu'en dessous de 80 Km/h, la vitesse est élevée à 0%. Elle est considérée comme élevée au dessus de 100 Km/h, elle est donc élevée à 100% au dessus de 100 Km/h

III.3.2 différentes formes des fonctions d'appartenance

La fonction d'appartenance μ comprise entre 0 et 1 est associée au sous ensemble E_1 de $T(x)$ et à la variable x , dont l'ensemble des valeurs possibles est $T(x)$. Elle s'appelle aussi, degré d'appartenance ou coefficient d'appartenance ou degré de possibilité, qui est la possibilité que la variable u ait la qualité associée au sous – ensemble E_1 . Elle est utilisée pour le traitement mathématique des variables linguistiques dans le but de traiter des déductions floues par ordinateur [18]. Ils existent plusieurs formes de la fonction d'appartenance dont ces fonctions sont les suivantes :

a) fonction triangulaire

$$\mu(x) = \begin{cases} \frac{x-a}{b-a} & \text{si } a < x \leq b \\ \frac{c-x}{c-b} & \text{si } b < x \leq c \\ 0 & \text{ailleurs} \end{cases} \quad (\text{III.1})$$

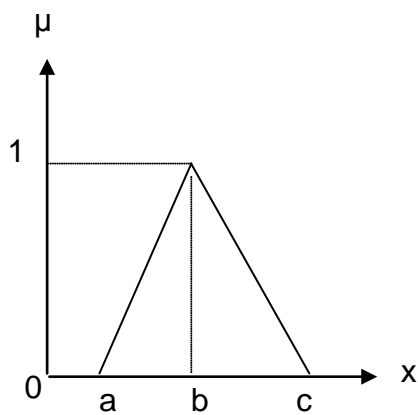
b) fonction d'appartenance trapézoïdale

$$\mu(x) = \begin{cases} \frac{x-a}{b-a} & \text{si } a < x \leq b \\ 1 & \text{si } b < x \leq c \\ \frac{d-x}{d-c} & \text{si } c < x \leq d \\ 0 & \text{ailleurs} \end{cases} \quad (\text{III.2})$$

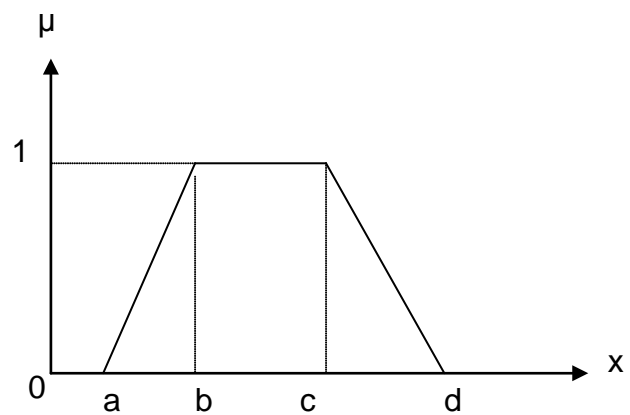
c) fonction d'appartenance gaussienne

$$\mu(x) = \exp \left[-\left(\frac{x-m}{\delta} \right)^2 \right] \quad -\infty < x \leq +\infty \quad (\text{III.3})$$

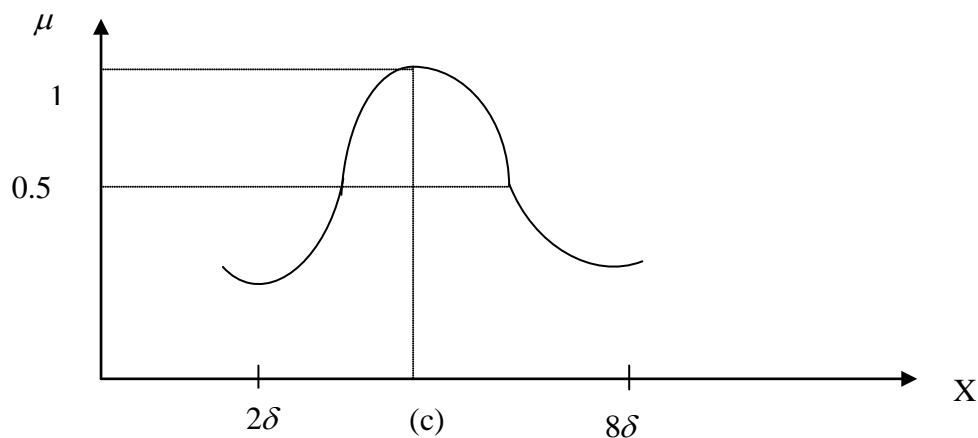
La figure III.4 représente les formes de ces trois types de fonction d'appartenance.



a) forme triangulaire



b) forme trapézoïdale



c) forme gaussienne

Fig. III.4 Différentes formes de la fonction d'appartenance

III.3.3 intervalles d'inférence

Un interval flou est une quantité floue convexe. Les intervalles flous définissent le nombre de variables floues associées à une grandeur d'entrée. Pour l'exemple de la vitesse, on aurait pu définir cinq intervalles. D'une façon générale, les sous-ensembles flous sont caractérisés à l'aide de symboles tels que ceux présentés dans le tableau.

Symbole	Signification
NG	Négatif Grand
NM	Négatif Moyen
EZ	Environ zéro
PM	Positif Moyen
PG	Positif Grand

Tab III.1 Symbole de cinq intervalles ou sous ensembles flous

Un nombre de valeurs linguistiques supérieures à sept n'apporte en générale aucune amélioration du comportement dynamique du réglage. Par contre, un tel choix compliquerait la fonction des règles d'inférences [19] [13] [20].

III.3.4 variable linguistique

Une variable linguistique est caractérisée par un triplet $(x, T(x), u)$ ou :

X : le nom de variable

T(x) : ensemble de terme de variable, c'est l'ensemble des noms des valeurs linguistiques de x dont chaque valeur est un sous ensemble flou défini dans U.

U : l'univers de discours

Comme exemple de la variable linguistique, on prend la vitesse de rotation du moteur. Son ensemble de terme T (vitesse) peut être :

T (vitesse) = (lente, moyenne, rapide, ...)

Ou chaque terme dans T (vitesse) est caractérisé par un ensemble flou dans l'univers de discours $U = [0,200]$ par exemple.

Pour les vitesses inférieures ou égales à 80 intervient la variable linguistique (lente), et pour les vitesses supérieures ou égales à 150 intervient la variable linguistique (rapide). Ces deux termes peuvent être caractérisés comme des ensembles flous dont les fonctions d'appartenance sont montrées sur la figure (III.5).

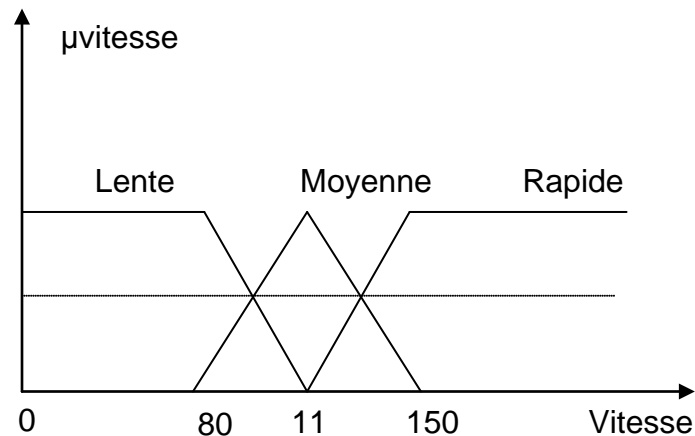


Figure III.5 Représentation des variables linguistiques

III.3.5 opérateurs de la logique floue

Puisque les fonctions caractéristiques des ensembles flous sont graduelles. L'extension de certains opérateurs définis sur les ensembles de la logique classique aux cas des ensembles flous pose certaines difficultés. Donc, il n'est possible d'appliquer directement la logique booléenne qui manipule des ensembles à fonction caractéristiques binaires (0 et 1). Pour pallier à ce problème, sont introduits quelques définitions d'outils nécessaires au traitement des ensembles flous.

Il existe plusieurs lois de composition des différents opérateurs de la logique floue, mais les plus couramment utilisées sont les suivants [15], [15], [13], [14] :

- **cas de l'opérateur ET** : il est réalisé par plusieurs méthodes tel que le maximum ou le produit des deux degrés d'appartenance $\mu(x_1)$ et $\mu(x_2)$ aux deux ensembles flous en question soit :

$$\mu(x_1, x_2) = \mu_{1i}(x_1) \mu_{2j}(x_2) \quad (\text{III.4})$$

- **cas de l'opérateur *OU*** : la réalisation de cet opérateur se fait par exemple par le calcul du maximum ou par la formation de la moyenne arithmétique des deux degrés d'appartenance $\mu_{1i}(x_1)$ et $\mu_{2j}(x_2)$ aux deux ensembles flous i et j de x_1 et x_2 respectivement, c'est-à-dire :

$$\mu(x_1, x_2) = \frac{1}{2}(\mu_{1i}(x_1) + \mu_{2j}(x_2)) \quad (\text{III.5})$$

- **cas de l'opérateur *ALORS*** : cet opérateur permet d'évaluer le degré de vérité des propositions floues, il est réalisé par plusieurs méthodes tel que le produit des deux fonctions d'appartenance $\mu_{1i}(x_1)$ et $\mu_{2j}(x_2)$ des deux ensembles flous i et j de x_1 et x_2 respectivement, c'est-à-dire :

$$\mu(x_1, x_2) = \mu_{1i}(x_1) \mu_{2j}(x_2) \quad (\text{III.6})$$

Les opérations minimum et maximum présentent l'avantage de la simplicité des calculs, par contre, elles privilégient l'une des deux variables. Les opérations de produit et valeur moyenne sont plus complexes à calculer mais elles produisent un résultat qui tient compte des valeurs de deux variables [15].

III.4 conception d'un contrôleur à logique floue

Les étapes de conception d'un contrôleur à logique floue sont les suivantes :

III.4.1 stratégie de fuzzification

La fuzzification est fondée sur la théorie des ensembles flous et joue un rôle essentiel pour la manipulation des données dans un contrôleur flou. Elle consiste à convertir les valeurs numériques à l'entrée du contrôleur en valeurs linguistique via des fonctions d'appartenances et ce, par une projection de ces valeurs dans l'univers de discours U en utilisant un facteur d'échelle k_e . On peut dire que la stratégie de fuzzification est un choix de l'univers de discours et des fonctions d'appartenance.

- *Univers de discours* : l'univers de discours représente l'intervalle de variation des grandeurs d'entrée du contrôleur flou après une normalisation par un facteur d'échelle.
- *Facteur d'échelle* : est le moyen de normalisation des grandeurs physiques dans une intervalle de $[1 \text{ } +1]$. Le facteur d'échelle doit être choisi sur la base de l'étude du système de sortie à condition que lors de petits phénomènes transitoire, le domaine admissible pour les grandeurs normalisées ne doivent pas être dépassés.

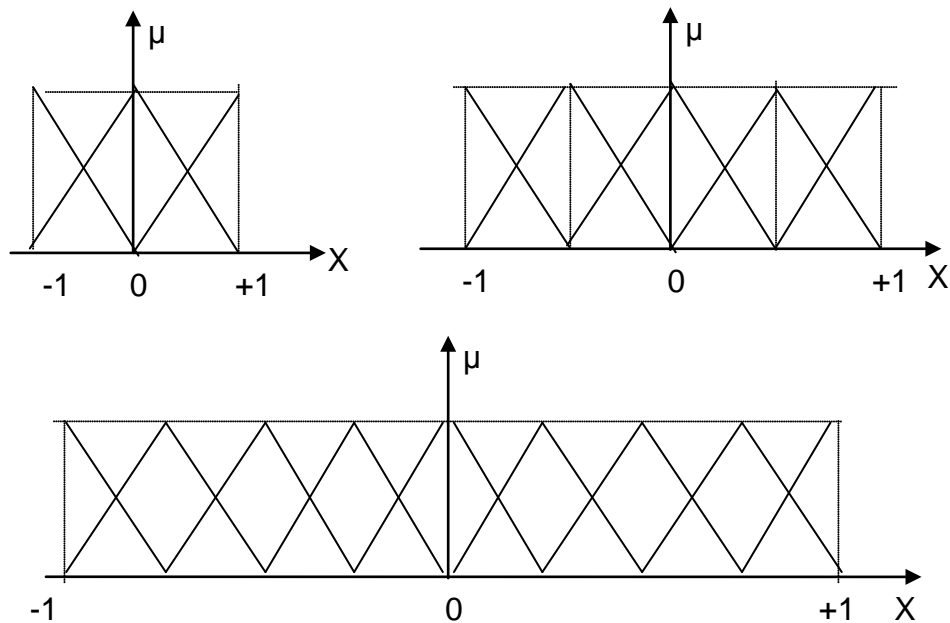


Figure III.6. Univers de discours normalisé dans l'intervalle $[-1 \ +1]$ avec une partition de 3, 5, 7 ensembles flous

III.4.2 base de données

- *Normalisation* : elle consiste à normaliser l'univers de discours U dans un intervalle bien déterminé.
- *Partition floue de l'espace d'entrée et de sortie* : c'est la division de l'univers de discours en plusieurs ensembles flous.

III.4.3 base de règles

Il existe quatre possibilités pour l'exploitation de la base de règles. La première est basée sur l'expertise humaine, la deuxième est fondée sur les actions de contrôle fournies par un opérateur, la troisième est l'utilisation du modèle flou du processus, la quatrième et la dernière qui est la plus récente, est basée sur un algorithme d'apprentissage pour l'élaboration des règles.

III.4.4 Inférence

Il existe deux types d'inférence :

- *Inférence à une seule règle* : c'est la comparaison de plusieurs concurrents dans une même situation et le résultat c'est le choix de l'optimum, ce type d'inférence n'est pas utilisé pour le contrôle flou.

- *Inférence à plusieurs règles* : dans ce type d'inférence, la prise de décision pour une ou plusieurs variables est différente pour chacun des variables, c'est l'inférence qui est utilisée dans les problèmes de réglage et de commande par logique floue. Les variables sont, soit des variables déterministes, ou des variables floues [16]. L'exemple de réglage de la température d'un locale est cité pour bien cerner ce type d'inférence.

La règle c'est la variation de la position de la valve ainsi l'objet de réglage c'est de minimiser l'écart. Pour se faire, les étapes suivantes dites règle d'inférences sont appliquées.

Si $e > 0$ alors $\Delta U > 0$ ou

Si $e < 0$ alors $\Delta U < 0$ ou

Si $e = 0$ alors $\Delta U = 0$

(III.7)

Avec ΔU : une variable floue

e : variable linguistique

Les décisions de la sortie ΔU sont données par l'expérience humaine, les règles d'inférence peuvent être représentées par la figure (III.7).

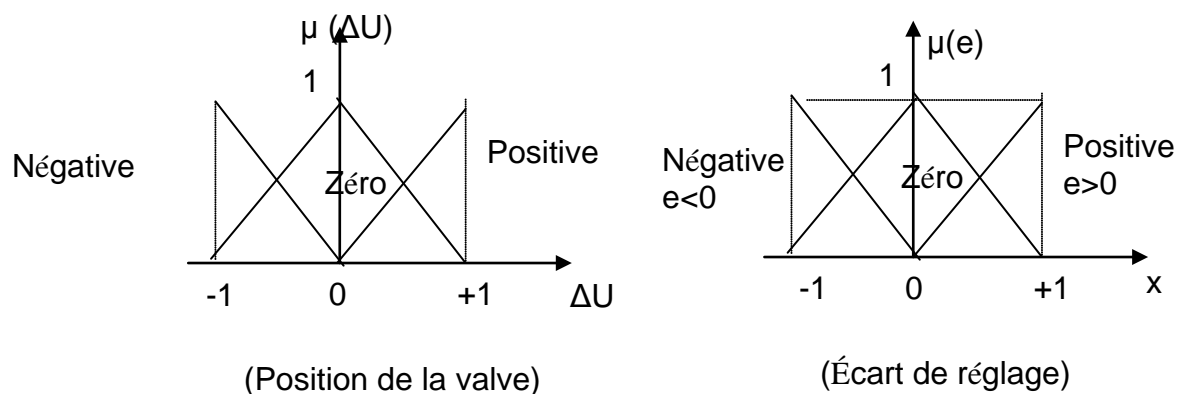


Figure III.7 Exemple d'inférence à plusieurs règles

- *Traitement numérique des inférences* :

Après avoir établi toutes les règles d'inférence on peut les représenter par un tableau ou par une matrice. Un traitement numérique des inférences selon l'une des trois méthodes suivantes à exécuter [21].

- méthode d'inférence max-min
- méthode d'inférence max-prod
- méthode d'inférence somme-prod

Considérons un ensemble de deux règles définies par :

R_1 : si x est A_1 et y est B_1 , alors Z est C_1 ou

R_2 : si x est A_2 et y est B_2 , alors Z est C_2

1. méthode d'inférence max-min

Cette méthode est appliquée au contrôleur dit de type " Mandani " [13]. Dans ce mode de raisonnement, la $i^{\text{ème}}$ règle aboutit à la décision de contrôle :

$$\mu_{R_i} = \mu_{R_M}(\alpha_i, \mu_{C_i}(Z)) = \min(\alpha_i, \mu_{C_i}(Z))$$

Avec : $\alpha_i = \mu_{R_M}(\mu_{A_i}(x_0), \mu_{B_i}(y_0)) = \min(\mu_{A_i}(x_0), \mu_{B_i}(y_0))$ qui représente la valeur de vérité des règles d'inférences.

Le résultat des deux règles est construit comme suit :

$$\mu_{Res}(Z) = \max(\mu_{R_1}(Z), \mu_{R_2}(Z)) \quad (III.8)$$

2. méthode d'inférence max-prod

C'est la méthode appliquée au contrôleur dit de type " Larsen " [22] cette méthode est basée sur l'utilisation du produit pour l'implication, dans ce cas la $i^{\text{ème}}$ règle donne la décision :

$$\mu_{R_i} = \alpha_i \cdot \mu_{C_i}(Z)$$

Avec : $\alpha_i = \mu_{R_M}(\mu_{A_i}(x_0), \mu_{B_i}(y_0)) = \min(\mu_{A_i}(x_0), \mu_{B_i}(y_0))$

La fonction d'appartenance résultante est donnée par :

$$\mu_{Res}(Z) = \max(\mu_{R_1}(Z), \mu_{R_2}(Z)) \quad (III.9)$$

3. méthode d'inférence somme-prod

Cette méthode est appliquée au contrôleur dit de type " Zadeh ". Elle définie comme suit :

$$\mu_{R_i} = \alpha_i \cdot \mu_{C_i}(Z)$$

$\alpha_i = \mu_{R_P}(\mu_{A_i}(x_0), \mu_{B_i}(y_0)) = \mu_{A_i}(x_0) + \mu_{B_i}(y_0)$

$$\mu_{Res}(Z) = \frac{1}{m} \sum_{i=1}^m \mu_{R_i} \quad (III.10)$$

Avec m : nombre de règles.

III.4.5 Stratégie de fuzzification

Pour bien définir la loi de commande, le contrôleur flou doit avoir une procédure de fuzzification jouant le rôle de la conversion de la commande floue en valeur physique pour chaque état du processus. Il existe plusieurs stratégies de défuzzification dont les plus utilisées sont [23] :

- méthode du maximum ;
- méthode du centre de gravité ;
- méthode des hauteurs pondérées ;

a- méthode du maximum

Pour cette méthode, la commande est égale à la commande locale ayant la fonction d'appartenance maximale :

$$U = \text{Max}_{z \in S} (\mu_{\text{Res}}(Z)) \quad (\text{III.11})$$

Où S est le support des μ_{Res}

b- méthode du centre de gravité

C'est la méthode la plus utilisée qui consiste à déterminer le centre de gravité de la fonction d'appartenance résultante $\mu_{\text{Res}}(Z)$.

Dans ce contexte il suffit de calculer l'abscisse U du centre de gravité comme suit :

$$U = \frac{\sum_{i=1}^n \mu_{\text{Res}}(Z_i) Z_i}{\sum_{i=1}^n \mu_{\text{ci}}(Z_i)} \quad (\text{III.12})$$

Où n est le nombre des niveaux de la sortie du contrôleur. Par cette méthode, on aboutit à une solution unique.

c- méthode des hauteurs pondérées

Cette méthode consiste à calculer la sortie du contrôleur en utilisant les centres de gravité Z^* de chacun des ensembles C_i

$$U = \frac{\sum_{i=1}^m \mu_{\text{ci}} \cdot Z_i^*(Z_i)}{\sum_{i=1}^m \mu_{\text{ci}}(Z_i)} \quad (\text{III.13})$$

Où m est le nombre d'ensembles flous associés à la variable de sortie U.

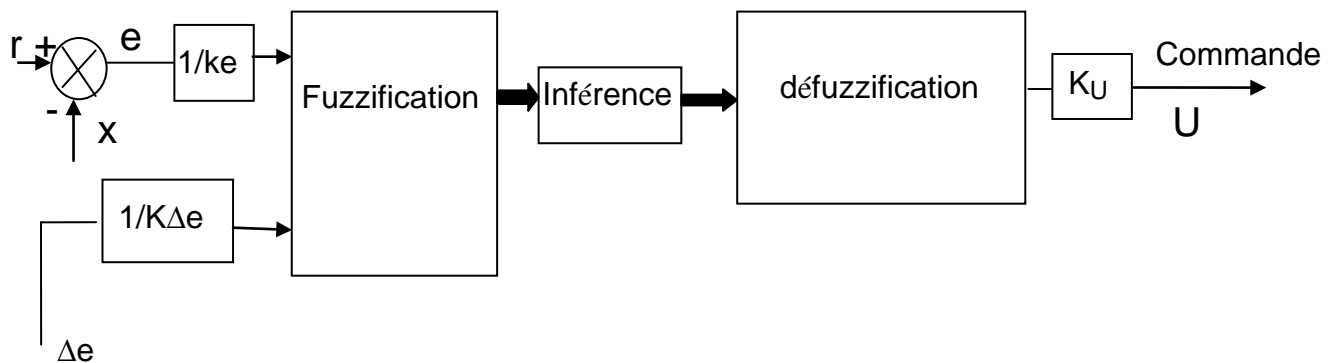
III.4.6 Structure du régulateur par logique floue

Le régulateur à logique floue possède en générale deux entrées, l'erreur "e" ($e = r - x$) et la variation de l'erreur Δe ($\Delta e = e - e_0$). Pour un régime transitoire convenable et un bon réglage de la vitesse, on utilise un facteur d'échelle K_e et $K_{\Delta e}$, K_u le domaine normalisé coïncide avec la variation maximale du signal de commande U_{com} . Pour cela on choisit une forme triangulaire pour la fonction d'appartenance distribuée de manière équidistante avec chevauchement ce qui conduit à une caractéristique linéaire ou quasi- linéaire [9].

Le bloc R.L.F est l'organe principale du régulateur contenant l'interface de fuzzification qui représente l'univers de discours et les variables linguistiques, puis l'inférence qui fournit l'action (la décision) de la commande, et l'inférence de défuzzification qui transforme la commande floue à une commande non floue pour contrôler notre système.

A la sortie du R.L.F, la variable de sortie est multipliée par un facteur d'échelle K_u pour fournir la variable normalisée U de la commande.

La configuration interne d'un contrôleur par logique floue est donnée par la figure (III.8).



III.8 Structure d'un régulateur à logique floue (RLF)

III.5 conception d'un contrôleur flou pour le réglage de la vitesse

Le choix des variables d'entrée et de sortie est la première étape à mener pour la conception d'un contrôleur flou. Les étapes suivantes : la normalisation de ces variables dans un domaine bien défini, la fuzzification, l'inférence et son traitement et finalement la défuzzification.

III.5.1 le choix des variables d'entrées et de sorties

Pour le réglage de la vitesse de la machine asynchrone, nous avons choisi comme variables d'entrées l'erreur 'e' qui est la différence entre la vitesse de référence Ω_{ref} imposée par l'opérateur, et la vitesse de rotation du rotor de la machine. L'autre variable d'entrée est la dérivée de l'erreur qui représente la différence entre l'erreur actuelle et l'erreur précédente ($\Delta e = e - e_0$). Comme variable de sortie une seule variable de commande qui est le couple de référence (C_{ref}) est choisie [14].

III.5.2 Normalisation des variables d'entrée et de sortie

La normalisation se fait par une division des variables $e, \Delta e$ et C_{ref} respectivement par les coefficients K_e et $K_{\Delta e}, K_u$ de telle sorte qu'elles évoluent dans le domaine $[-1 \ 1]$ pour toutes les variables.

III.5.3 Interface de fuzzification

La fuzzification représente d'une part le choix de l'univers de discours des variables linguistique. Ce choix est généralement basé sur l'expérience de l'opérateur. L'intervalle de l'univers de discours de l'erreur est choisi entre $[-0.1 \ 0.1]$, pour la dérivée de l'erreur entre $[-0.1 \ 0.1]$ et pour la sortie entre $[-0.1 \ 0.1]$.

D'autre part la fuzzification consiste à choisir la forme des fonctions d'appartenance. Pour une raison de simplification en vue d'avoir le même effet de réglage dans les sens de rotation, les fonctions d'appartenance de forme triangulaire avec intersection de 50 % placées symétriquement par rapport à zéro sont utilisées. Le nombre des ensembles flous est sept pour toutes les variables linguistiques qui sont successivement noté :

NG : Négatif Grand ; **NM** : Négatif Moyen ; **NP** : Négatif Petit ; **EZ** : Environ Zéro ; **PP** : Positif Petit ; **PM** : Positif Moyen ; **PG** : Positif Grand.

La représentation de ces ensembles flous est donnée par la figure III.9.

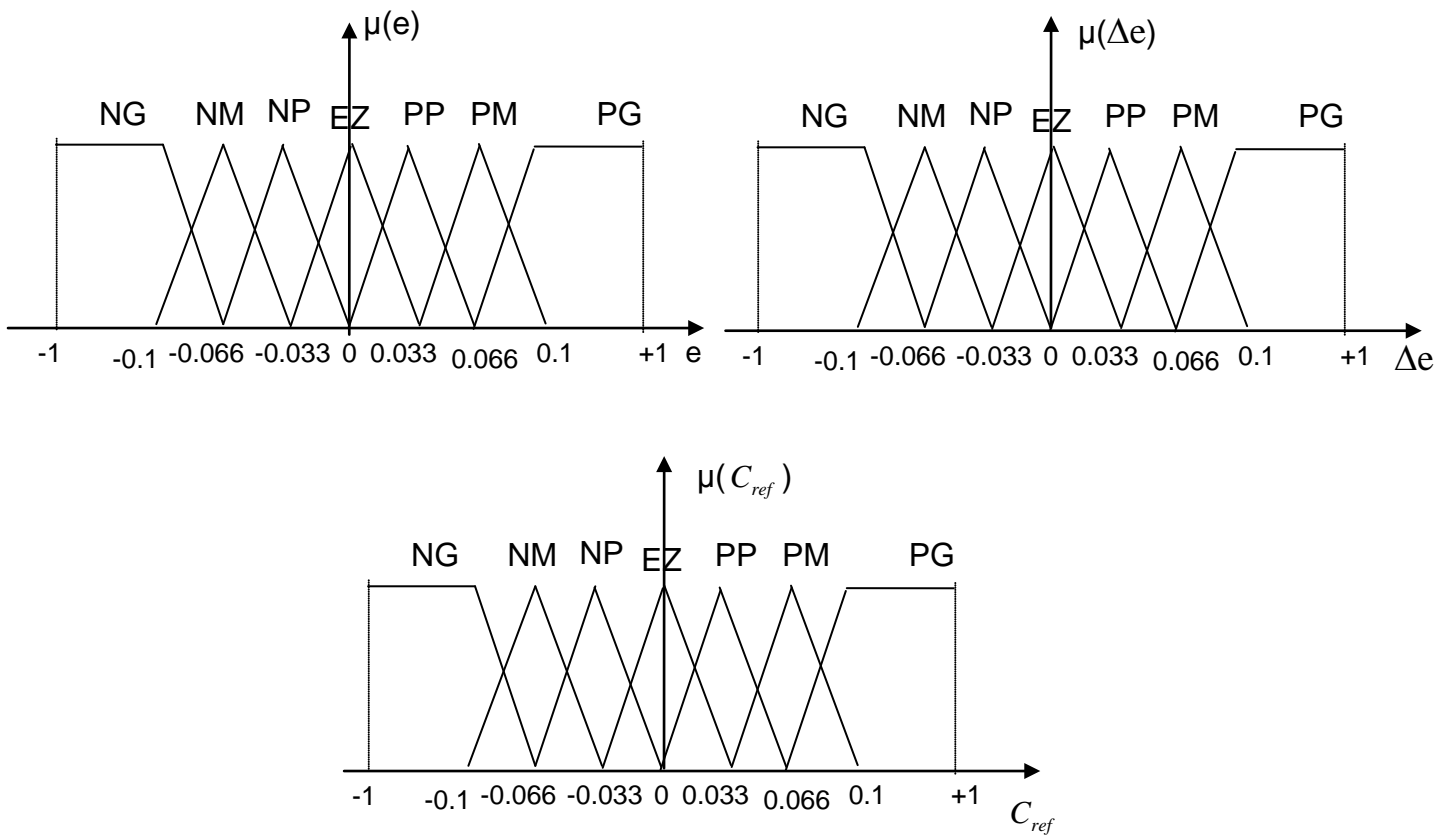


Figure III.9 Description des fonctions d'appartenance

III.5.4 Inférence

L'inférence ou la prise de décision est le noyau du contrôleur flou, elle est capable de simuler la prise de décision de l'être humain en se basant sur les concepts flous et les règles d'inférence, les règles d'inférence utilisées sont de type :

Si e est x_1 ET Δe est x_2 ALORS C_{ref} est x_3

Le nombre des ensembles flous pour e et Δe est sept donc il faut $7 \times 7 = 49$ règles d'inférence. Sachant qu'il n'existe pas une loi bien déterminée pour la détermination de la décision de chaque règle. L'expérience humaine, dans ce cas joue un rôle très important.

Les règles d'inférence pour notre système sont représentées par une matrice d'inférence selon le tableau III.1

III.5.5 Traitement numérique de l'inférence

Après avoir choisi les règles d'inférence, les opérateurs de la logique floue permettent de choisir une méthode pour le traitement de l'inférence. Sachant que l'opérateur *ET* est le minimum, l'opérateur *OU* est le maximum et l'opérateur *ALORS* est le maximum. La méthode retenue sera la méthode min/max.

$\Delta e/e$	NG	NM	NP	EZ	PP	PM	PG
NG	NG	NM	NM	NP	EZ	EZ	EZ
NM	NG	NM	NM	NP	EZ	EZ	EZ
NP	NG	NP	NP	EZ	EZ	EZ	PP
EZ	NM	NP	NP	EZ	PP	PP	PM
PP	NP	EZ	EZ	EZ	PP	PP	PG
PM	EZ	EZ	EZ	PP	PM	PM	PG
PG	EZ	EZ	EZ	PP	PM	PM	PG

Tab III.2. Matrice d'inférence de 7 ensembles flous

III.5.6 Défuzzification

La défuzzification consiste à transformer la décision de sortie linguistique en une décision numérique connue pour commander le système. Pour notre cas d'étude nous avons utilisé la méthode du centre de gravité.

III.5.6.1 Etude des différents sous ensembles

En ce qui concerne notre travail, nous avons retenu trois cas d'études, et ce, à base de l'étude théorique présentée dès le début de ce chapitre ; nous présentons l'analyse et la simulation des trois cas dans le but de valider la commande et estimer les performances de la vitesse.

1^{ère} cas : trois ensembles flous

Dans ce premier cas, on attribue à l'univers de discours des variables, trois ensembles flous : Négative (N), Zéro (Z) et Positif (P).

Par conséquent, la matrice d'inférence qui aura 9 règles est la suivante :

$\Delta e/e$	N	Z	P
N	N	N	Z
Z	N	Z	P
P	Z	P	P

Tab III.3 Matrice d'inférence de 3 ensembles flous

2^{ème} cas : cinq ensembles flous

Dans ce cas on subdivise l'univers de discours en cinq ensembles flous (GN, N, Z, P, et GP), la matrice d'inférence sera la suivante :

$\Delta e/e$	GN	N	Z	P	GP
GN	GN	GN	N	N	Z
N	GN	N	N	Z	P
Z	N	N	Z	P	P
P	N	Z	P	P	GP
GP	Z	P	P	GP	GP

Tab III.4. Matrice d'inférence de 5 ensembles flous

3^{ème} cas : sept ensemble flous

On attribue aux entrées sept ensembles flous (GN, MN, PN, EZ, PP, MP, GP). la matrice d'inférence sera la suivante.

$\Delta e/e$	NG	NM	NP	EZ	PP	PM	PG
NG	NG	NM	NM	NP	EZ	EZ	EZ
NM	NG	NM	NM	NP	EZ	EZ	EZ
NP	NG	NP	NP	EZ	EZ	EZ	PP
EZ	NM	NP	NP	EZ	PP	PP	PM
PP	NP	EZ	EZ	EZ	PP	PP	PG
PM	EZ	EZ	EZ	PP	PM	PM	PG
PG	EZ	EZ	EZ	PP	PM	PM	PG

III.6 Simulation et interprétation

La simulation des comportements de la vitesse et du couple qui est effectuée sous cette condition (trois ensembles) est représentée par la figure III.10.a. La vitesse est plus rapide et se stabilise à sa valeur de consigne à l'instant 0.2 seconde. Le couple électromagnétique résultant est d'abord limité au démarrage à la valeur 15 Nm, puis décroît progressivement ou il prendra la valeur nulle dès que la vitesse se stabilise. Il est clair que plus on subdivise de disques avec plus d'ensemble flous, plus le choix des règles devient plus vaste et son optimisation dépend du système à régler.

Les figures III.11, représentent le comportement du couple et de la vitesse tenant compte d'un univers de disques à cinq ensembles. On constate, que l'évolution de la vitesse en régime transitoire est plus rapide.

Les résultats de simulation des figures (III.14, III.15, III.16, III.17, III.18) illustrent l'évolution de la vitesse de rotation, du couple, des flux rotoriques et des courants statoriques en présence et absence des variations des paramètres de système. On présente d'abord un démarrage optimal présenté par la figure (III.14), ensuite application d'une charge à $t=0.5$ s (III.15), la vitesse atteint sa valeur de référence, les flux gardent leurs orientations et le couple électromagnétique suit le couple de charge et de référence. Une simulation d'un démarrage lors de l'augmentation du moment d'inertie de trois fois de la valeur nominale a été faite (figure III.17). La vitesse se stabilise en un temps plus court. Enfin les figures (III.18), montrent les effets de l'augmentation de la résistance rotorique de 50% de la valeur nominale. L'influence de ce type de variation est presque non perceptible.

III.7. Conclusion

La technique de réglage par logique floue, permet à la commande d'être insensible à de telles variations, la vitesse est peu affectée par les perturbations et suit sa consigne. On peut conclure que la commande par logique floue associée à la commande vectorielle se prête bien pour le réglage de la vitesse de la machine asynchrone. Elle assure une bonne orientation du flux rotorique et donc un bon découplage entre le flux et le couple.

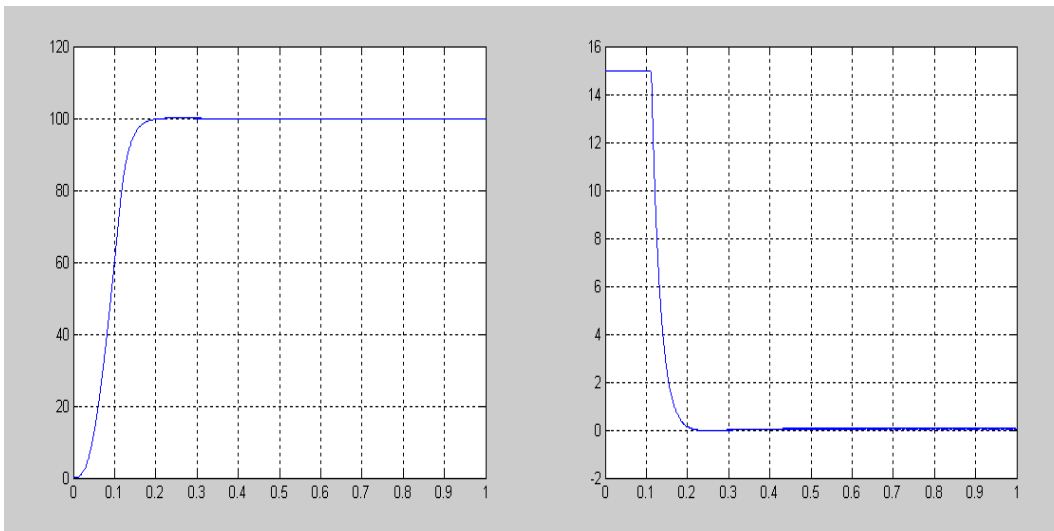


Figure III.10 Vitesse et couple de 3 sous ensembles

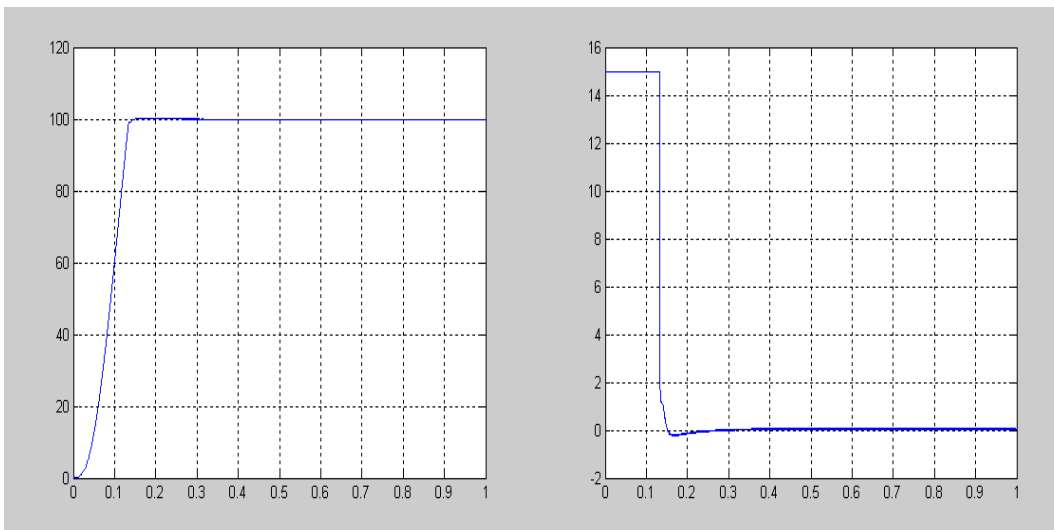


Figure III.11 Vitesse et couple de 5 sous ensembles

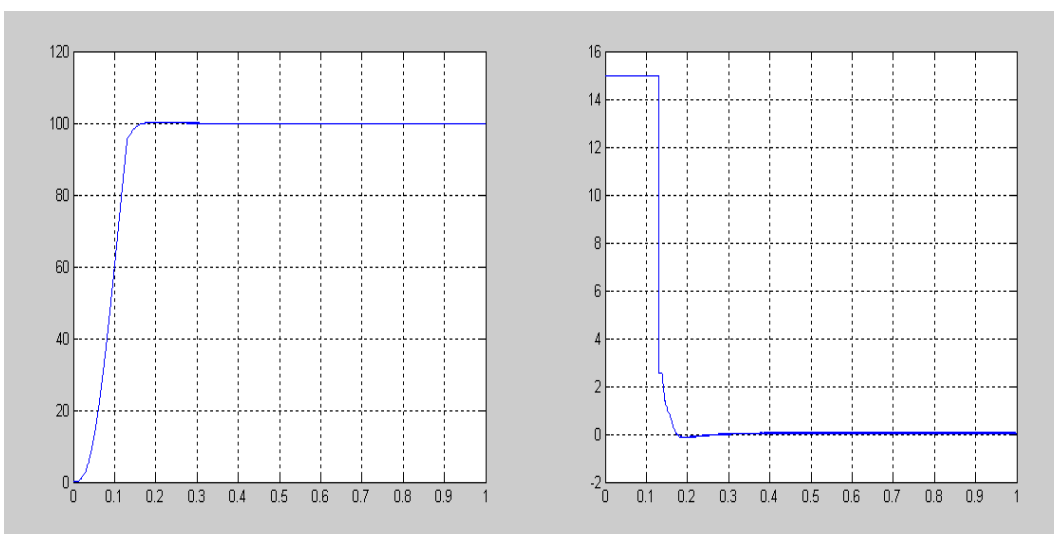


Figure III.12 Vitesse et couple de 7 sous ensembles

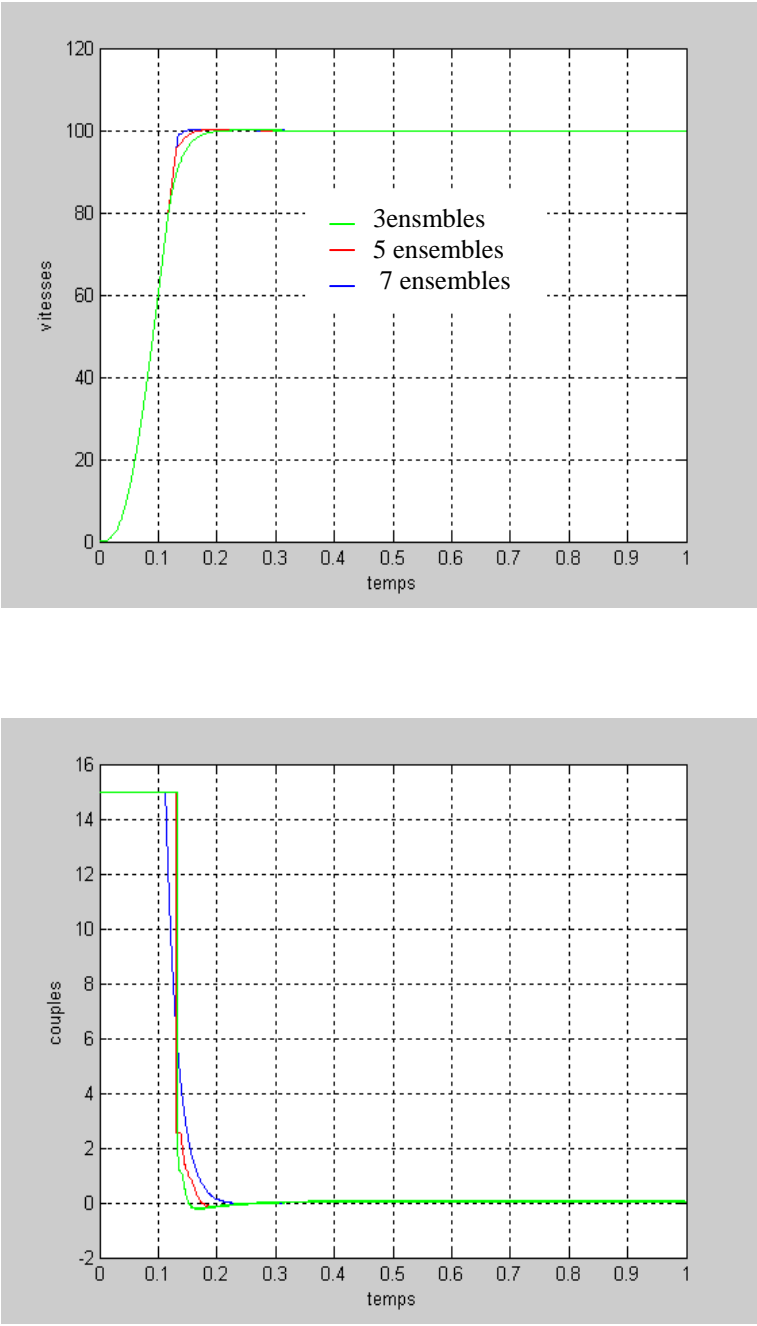


Figure III.13 Comparaison des sous ensembles 3,5 et 7

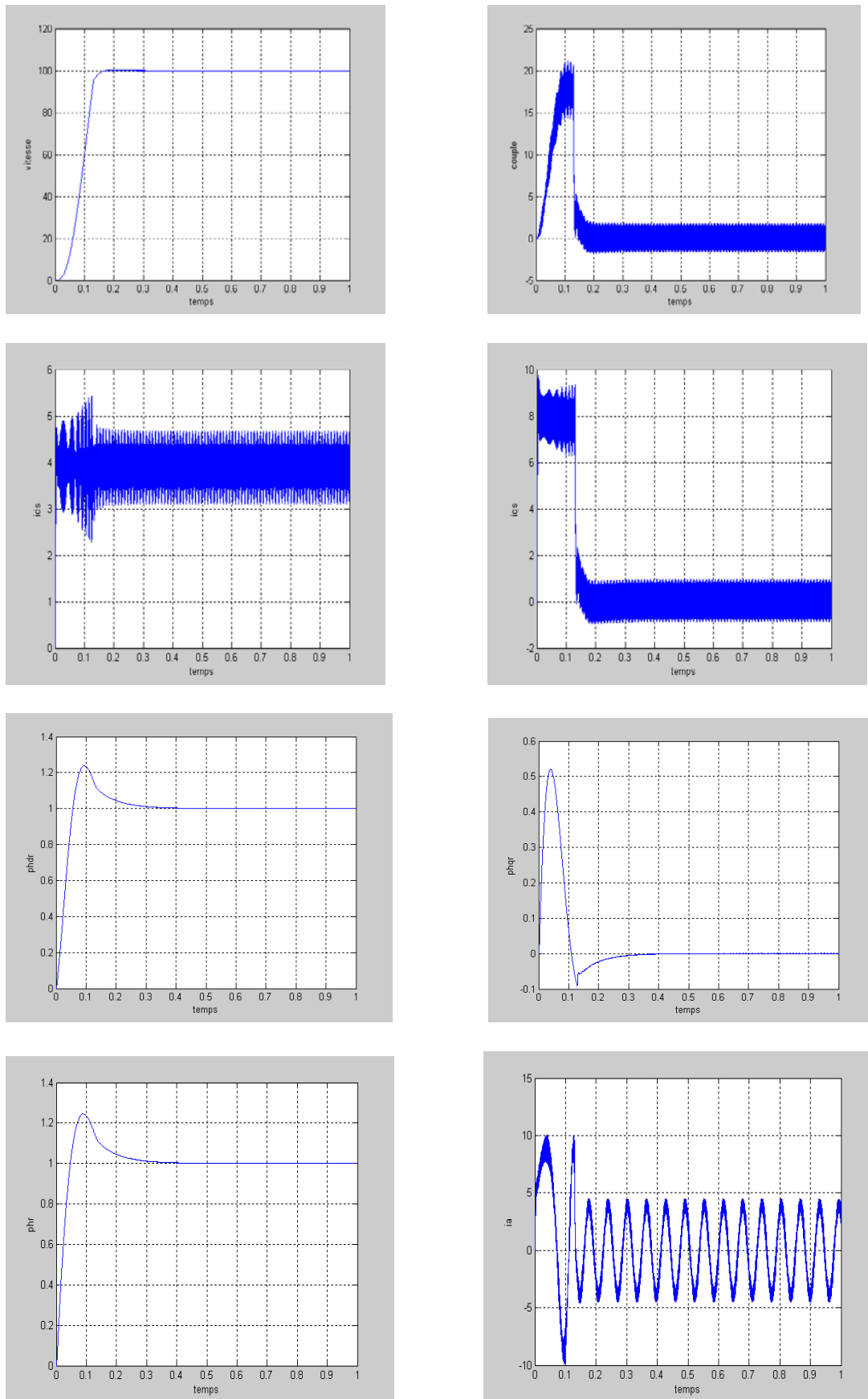


Figure III.14 RLF de la vitesse lors d'un démarrage à vide pour une vitesse de 100 rd/s

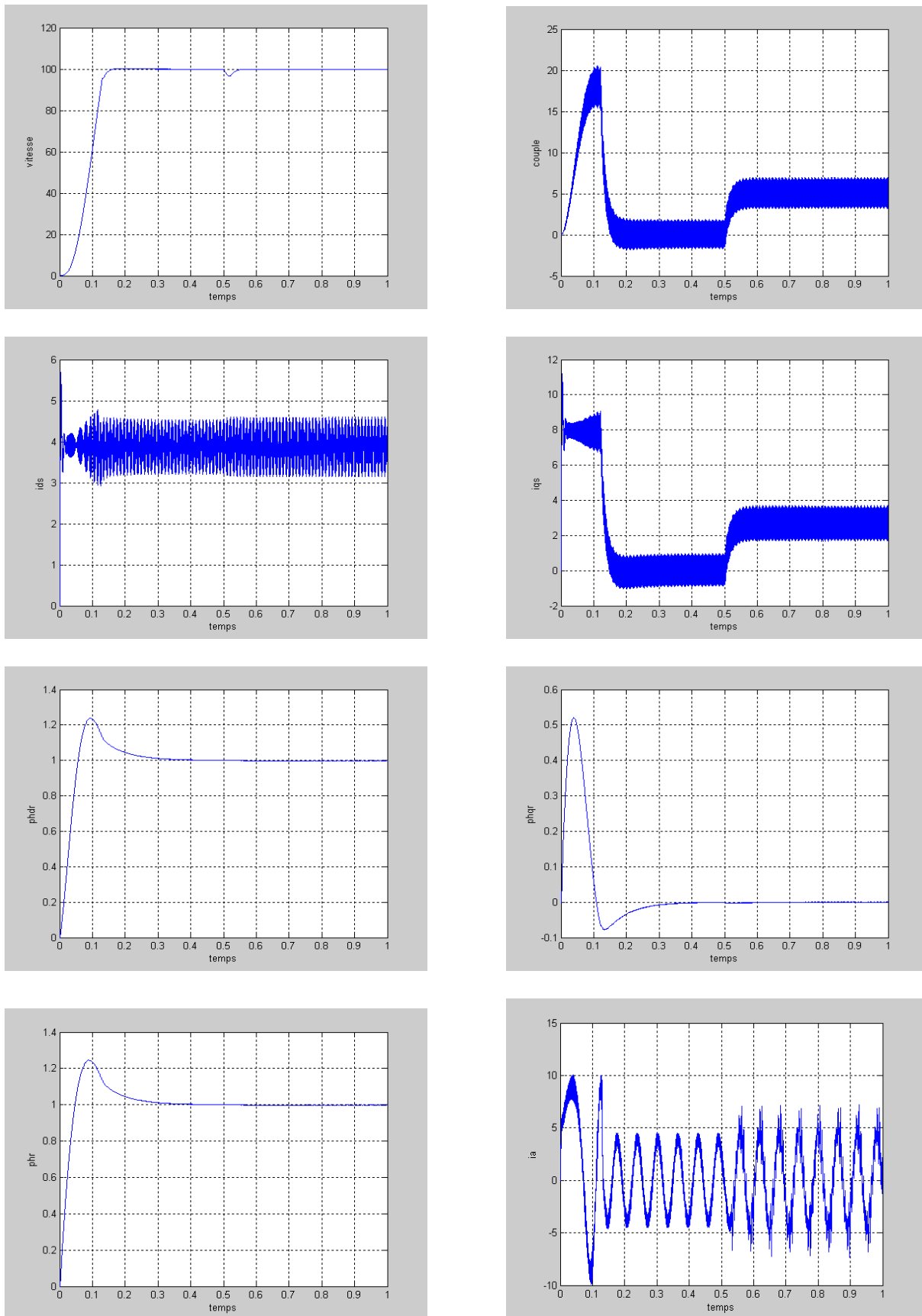


Figure III.15 RLF de la vitesse avec application d'un couple de charge à $t=0.5$ s

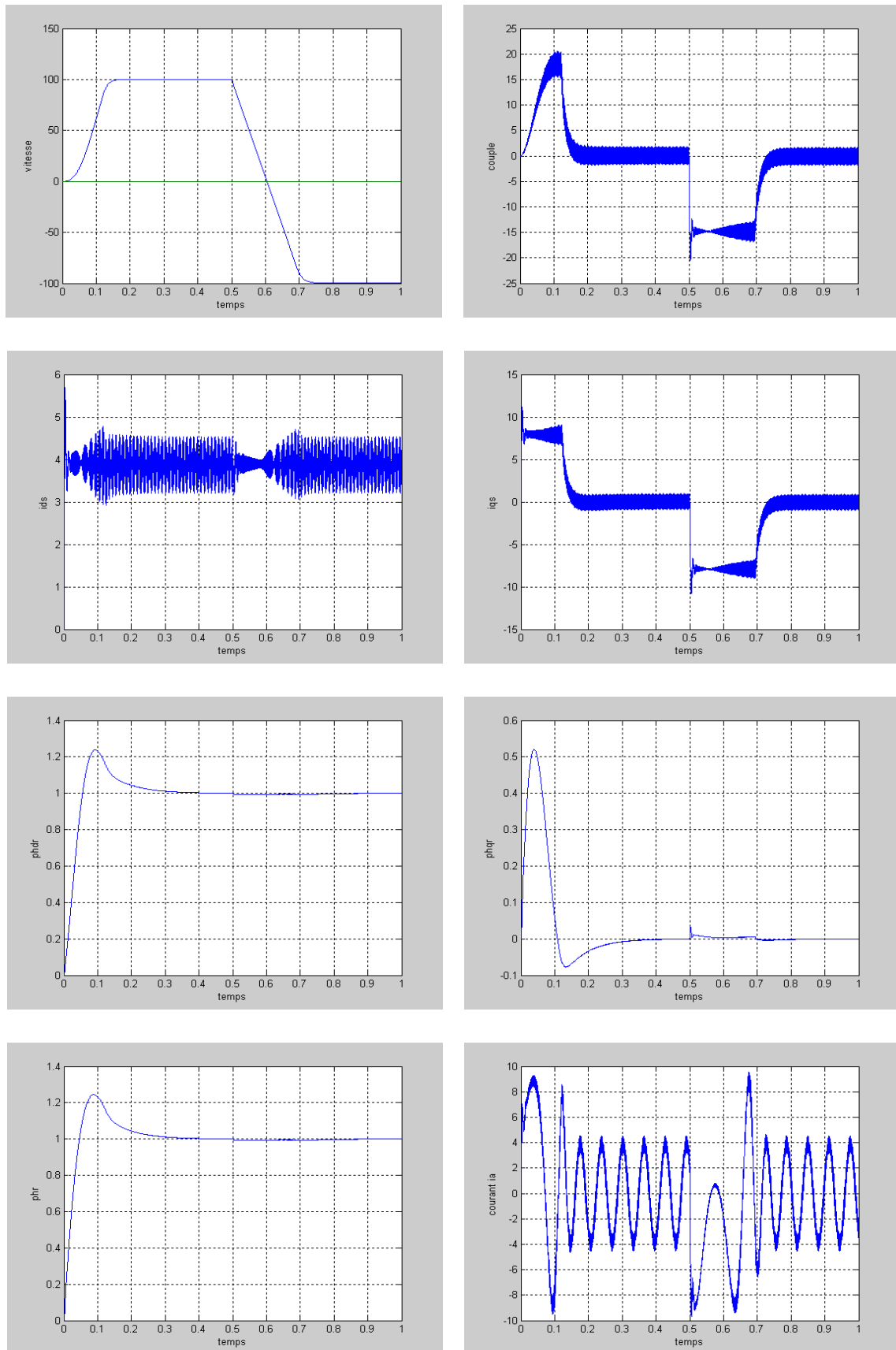


Figure III.16 RLF de la vitesse avec inversion de sens de rotation à $t=0.5$ s

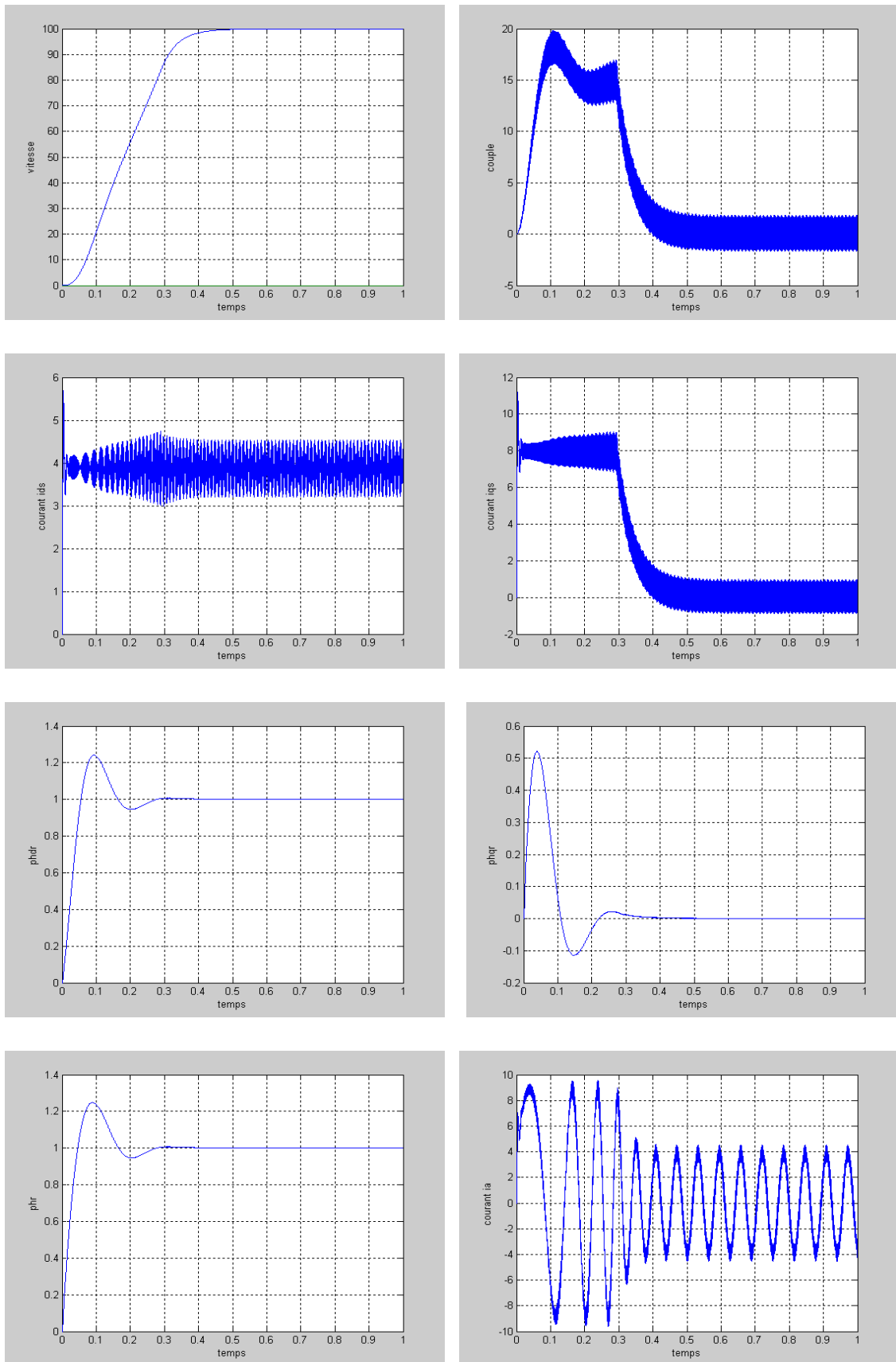


Figure III.17 RLF de la vitesse avec augmentation du moment d'inertie de $3 * J_n$

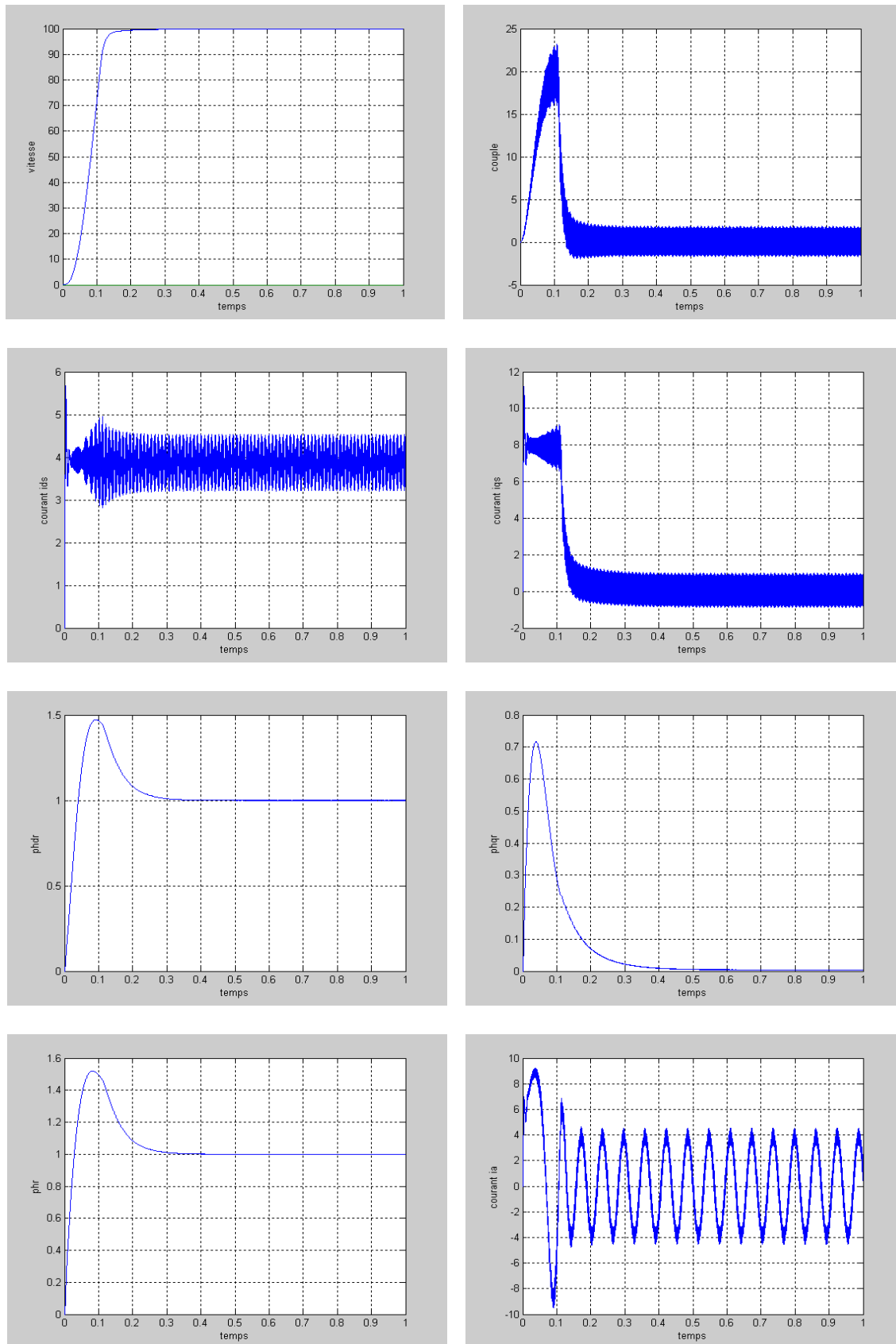


Figure III.18 RLF de la vitesse avec augmentation de la résistance rotorique de 50% R_{rn}

SYNTHESE DE REGULATEUR NEURONE

IV.1 Historique des réseaux de neurones

Brièvement, les premières recherches remontent à la fin 19e et au début du 20e siècle. Elles consistent en de travaux multidisciplinaires en physique, en psychologie et en neurophysiologie par des scientifiques tels Hermann von Helmholtz, Ernst Mach et Ivan Pavlov. `A cette époque, il s'agissait de théories plutôt générales sans modèle mathématique précis d'un neurone. On s'entend pour dire que la naissance du domaine des réseaux de neurones artificiels remonte aux années 1940 avec les travaux de Warren McCulloch et Walter Pitts qui ont montré qu'avec de tels réseaux, on pouvait, en principe, calculer n'importe quelle fonction arithmétique ou logique. Vers la fin des années 1940, Donald Hebb a ensuite proposé une théorie fondamentale pour l'apprentissage [24].

La première application concrète des réseaux de neurones artificiels est survenue vers la fin des années 1950 avec l'invention du réseau dit «perceptron» par un dénommé Frank Rosenblatt. Rosenblatt et ses collègues ont construit un réseau et démontré ses habilités `a reconnaître des formes. Malheureusement, il a été démontré par la suite que ce perceptron simple ne pouvait résoudre qu'une classe limitée de problème. Environ au même moment, Bernard Widrow et Ted Hoff ont proposé un nouvel algorithme d'apprentissage pour entraîner un réseau adaptatif de neurones linéaires, dont la structure et les capacités sont similaires au perceptron.

Vers la fin des années 1960, un livre publié par Marvin Minsky et Seymour Papert est venu jeter beaucoup d'ombre sur le domaine des réseaux de neurones. Entre autres choses, ces deux auteurs ont démontré les limitations des réseaux développés par Rosenblatt et Widrow Hoff. Beaucoup de gens ont été influencés par cette démonstration qu'ils ont généralement mal interprétée. Heureusement, certains chercheurs ont persévéré en développant de nouvelles architectures et de nouveaux algorithmes plus puissants. En 1972, Teuvo Kohonen et James Anderson ont développé indépendamment et simultanément de nouveaux réseaux pouvant servir de mémoires associatives. Également, Stephen Grossberg a investigué ce qu'on appelle les réseaux auto-organisés.

Dans les années 1980, une pierre d'achoppement a été levée par l'invention de l'algorithme de Rétro propagation des erreurs. Cet algorithme est la réponse aux critiques de Minsky et Papert formulées à la fin des années 1960. C'est ce nouveau développement, généralement attribué à David Rumelhart et James McClelland, mais aussi découvert plus ou moins en même temps par Paul Werbos et par Yann LeCun, qui a littéralement ressuscité le domaine des réseaux de neurones. Depuis ce temps, c'est un domaine où bouillonnent constamment de nouvelles théories, de nouvelles structures et de nouveaux algorithmes.

IV.2 Application

Les réseaux de neurones servent aujourd'hui à toutes sortes d'applications dans divers domaines. Par exemple, on a développé un auto-pilote pour avion, ou encore un système de guidage pour automobile, on a conçu des systèmes de lecture automatique de chèques bancaires et d'adresses postales, on produit des systèmes de traitement du signal pour différentes applications militaires, un système pour la synthèse de la parole, des réseaux sont aussi utilisés pour bâtir des systèmes de vision par ordinateur, pour faire des prévisions sur les marchés monétaires, pour évaluer le risque financier ou en assurance, pour différents processus manufacturiers, pour le diagnostic médical, pour l'exploration pétrolière ou gazière, en robotique, en télécommunication [24].

IV.3 Définition du neurone formel

Une définition très générale : processeur qui applique une opération simple a ses entrées et que l'on peut relier a d'autres pour former un réseau qui peut réaliser une relation entrée sortie quelconque.

Définition usuelle : processeur très simple qui calcule une somme pondérée et qui applique à cette somme une fonction de transfert non linéaire (échelon, sigmoïde, gaussienne, ...) [25]

IV.4 Elément de base des réseaux de neurones

L'élément de base d'un réseau de neurones est, bien entendu, le neurone artificiel. Un neurone contient deux éléments principaux:

- Un ensemble de poids associés aux connexions du neurone, et
- Une fonction d'activation.

Les valeurs d'entrée sont multipliées par leur poids correspondant et additionnées pour obtenir la somme S . [26]

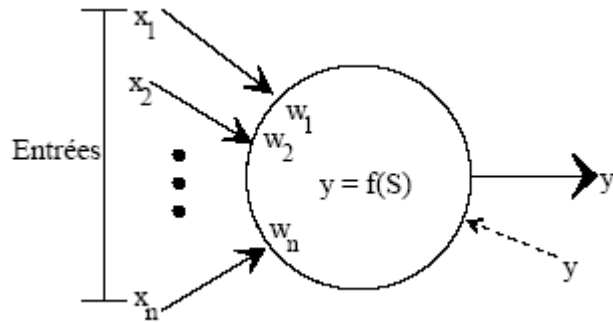


Fig. IV.1 le neurone artificiel

Les entrées du neurone sont désignées par $x_i (i = 1..n)$. Les paramètres w_n reliant les entrées aux neurones sont appelés *poids synaptique* ou tout simplement poids. La sortie du neurone est donnée par :

$$\text{sortie} = f(x) \quad \text{Avec} \quad x = \sum_{i=1}^n w_i x_i \quad (\text{IV.1})$$

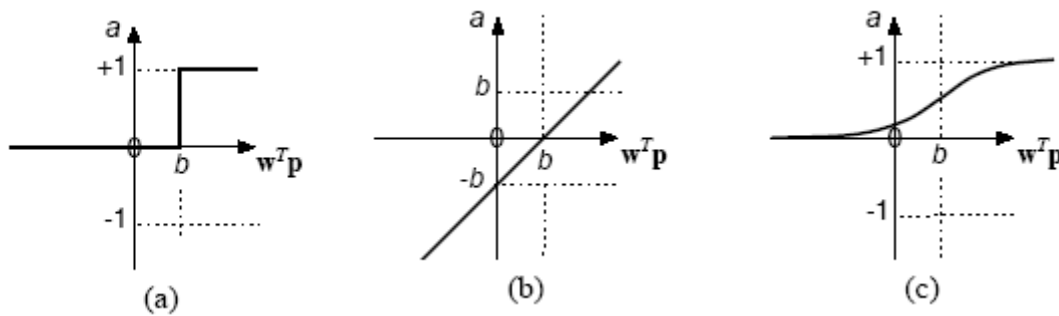
x est l'état d'activation du neurone (également appelé état ou activité)

f est la fonction d'activation du neurone, les fonctions d'activation sont généralement croissantes et bornées. Les fonctions les plus connues sont la fonction signe, la fonction linéaire et la fonction sigmoïde.

Les réseaux de neurone sont constitués des neurones élémentaires connectés entre eux par l'intermédiaire des poids qui jouent le rôle des synapses. L'information est portée par la valeur de ces poids tandis que la structure de réseau de neurone ne sert qu'à traiter cette information et à l'acheminer vers la sortie. Cela signifie que ses agents (neurones) s'organisent et modifient leurs liens mutuels lors d'une procédure fondamentale qu'est *l'apprentissage*. Pour une tâche précise, l'apprentissage du réseau de neurones consiste donc à adapter les différents poids w_i [16].

IV.5 Fonctions de transfert

Jusqu'à présent, nous n'avons pas spécifié la nature de la fonction d'activation de notre modèle. Il se trouve que plusieurs possibilités existent. Différentes fonctions de transfert pouvant être utilisées comme fonction d'activation du neurone sont énumérées au tableau 2.1. Les trois les plus utilisées sont les fonctions «seuil» (en anglais «hard limit»), «linéaire» et «sigmoïde». Comme son nom l'indique, la fonction seuil applique un seuil sur son entrée. Plus précisément, une entrée négative ne passe pas le seuil, la fonction retourne alors la valeur 0 (on peut interpréter ce 0 comme signifiant *faux*). [24]












(a) neurone seuil

(b) neurone linéaire

(c) neurone sigmoïde

Fig. IV.2 Fonctions de transfert

Nom de la fonction	Relation d'entrée/sortie	Icône	Nom Matlab
seuil	$a = 0$ si $n < 0$ $a = 1$ si $n \geq 0$		hardlim
seuil symétrique	$a = -1$ si $n < 0$ $a = 1$ si $n \geq 0$		hardlims
linéaire	$a = n$		purelin
linéaire saturée	$a = 0$ si $n < 0$ $a = n$ si $0 \leq n \leq 1$ $a = 1$ si $n > 1$		satlin
linéaire saturée symétrique	$a = -1$ si $n < -1$ $a = n$ si $-1 \leq n \leq 1$ $a = 1$ si $n > 1$		satlins
linéaire positive	$a = 0$ si $n < 0$ $a = n$ si $n \geq 0$		poslin
sigmoïde	$a = \frac{1}{1+\exp^{-n}}$		logsig
tangente hyperbolique	$a = \frac{e^n - e^{-n}}{e^n + e^{-n}}$		tansig
compétitive	$a = 1$ si n maximum $a = 0$ autrement		compet

Tab IV.1 fonction de transfert $a=f(n)$

VI.6 Réseaux de neurone

Un neurone étant une fonction non linéaire paramétrée, un réseau de neurones réalise une combinaison, elle-même paramétrée, de telles fonctions. On coutume de représenter cette combinaison sous la forme de neurones, comme représenté sur la Figure 1, reliés entre eux par des connexions qui représentent les poids. On distingue conventionnellement deux types de neurones dans un réseau [27] :

- *Les neurones cachés*, caractérisés en ce que leurs sorties ne constituent pas la sorties du réseau, mais sont combinées par le (ou les) neurone(s) de sortie pour constituer celle(s)-ci : ils sont dits cachés parce que leur sortie n'est pas une sortie du réseau.
- *Les neurones de sortie*, combinent les sorties des neurones cachés pour constituer les sorties du réseau. Pour des réseaux destinés à la modélisation,

Les neurones de sortie sont généralement des "neurones linéaires" (leur fonction d'activation est l'identité) : ainsi, la sortie d'un réseau de neurones destiné à la modélisation statique de processus est une combinaison linéaire paramétrée des fonctions non linéaires paramétrées des variables.

[28]

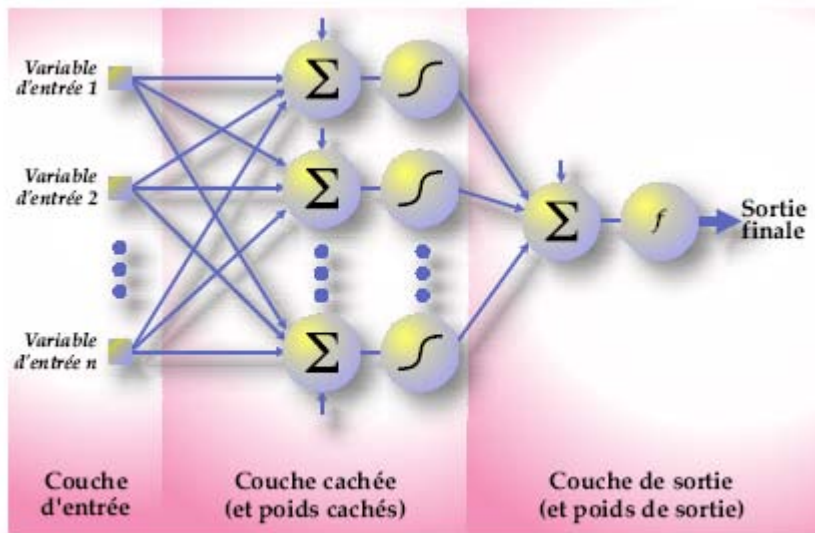


Fig. IV.3 topologie des réseaux de neurone

Dans chaque unité de la couche cachée les variables sont combinées de façon linéaire. Le réseau de neurones applique une transformation non linéaire à chacune de ces combinaisons.

Finalement, les valeurs résultantes des unités cachées sont combinées de façon linéaire pour obtenir la valeur prédite [29].

IV.6.1 Choix du nombre d'unités cachées

Le nombre d'unités cachées joue un rôle crucial dans le contrôle de la capacité du réseau de neurones. Si la valeur de n_h (est le nombre d'unités cachées du réseau) est trop petite, alors le réseau possède trop peu de paramètres et ne peut capter toutes les dépendances qui servent à modéliser et prédire les valeurs du processus observé. À l'inverse, si l'on choisit une valeur trop grande pour n_h , alors le nombre de paramètres du modèle augmente et il devient possible, pendant la phase d'optimisation des paramètres, de modéliser certaines relations qui ne sont que le fruit de fluctuations statistiques propres à l'ensemble d'entraînement utilisé plutôt que des relations fondamentales de dépendance entre les variables. Il faut comprendre que les réseaux de neurones sont des approximateurs universels, c.-à-d. qu'ils peuvent modéliser n'importe quelle fonction si le nombre d'unités cachées est suffisant. Autrement dit, un réseau de neurones peut apprendre par coeur un ensemble d'entraînement. Afin de s'assurer que le réseau de neurones s'en tient aux relations fondamentales de dépendance, on utilise, en plus de l'ensemble d'entraînement, un second ensemble appelé ensemble de validation : à la fin de chaque époque d'entraînement, on mesure non seulement l'erreur d'entraînement mais aussi l'erreur de validation, c.-à-d. l'erreur totale commise sur tous les exemples de l'ensemble de validation. Cette erreur de validation est calculée une fois que la phase d'optimisation des paramètres est terminée. Après avoir entraîné quelques modèles, chacun avec un nombre différent d'unités cachées, on peut comparer les erreurs d'entraînement et de validation. On obtient généralement le résultat suivant : l'erreur d'entraînement diminue au fur et à mesure que le nombre d'unités cachées augmente. L'erreur de validation, quant à elle, est élevée lorsque le nombre d'unités cachées est faible, décroît avec l'augmentation du nombre d'unités cachées, atteint un minimum pour un certain nombre optimal d'unités cachées, puis croît lorsque le nombre d'unités devient trop grand. C'est donc l'utilisation d'un ensemble de validation, distinct de l'ensemble d'entraînement, qui nous permet de choisir le nombre optimal d'unités cachées ou neurones [27].

IV.6.2 Propriétés de réseaux de neurones

Un réseau de neurones se compose de neurones qui sont interconnectés de façon à ce que la sortie d'un neurone puisse être l'entrée d'un ou plusieurs autres neurones. Ensuite il y a des

entrées de l'extérieur et des sorties vers l'extérieur [30]. Rumelhart et al. Donnent huit composants principaux d'un réseau de neurones :

- Un *ensemble de neurones*
- Un *état d'activation* pour chaque neurone (actif, inactif, ...)
- Une *fonction de sortie* pour chaque neurone ($f(S)$)
- Un *modèle de connectivité* entre les neurones (chaque neurone est connecté à tous les autres, par exemple)
- Une *règle de propagation* pour propager les valeurs d'entrée à travers le réseau vers les sorties
- Une *règle d'activation* pour combiner les entrées d'un neurone (très souvent une somme pondérée)
- Une *règle d'apprentissage*
- Un *environnement* d'opération (le système d'exploitation, par exemple)

Le comportement d'un réseau et les possibilités d'application dépendent complètement de ces huit facteurs et le changement d'un seul d'entre eux peut changer le comportement du réseau complètement.

Les réseaux de neurones sont souvent appelés des "*boîtes noires*" car la fonction mathématique qui est représentée devient vite trop complexe pour l'analyser et la comprendre directement. Cela est notamment le cas si le réseau développe des représentations distribuées [30], c'est à dire que plusieurs neurones sont plus ou moins actifs et contribuent à une décision. Une autre possibilité est d'avoir des représentations localisées, ce qui permet d'identifier le rôle de chaque neurone plus facilement. Les réseaux de neurones ont quand même une tendance à produire des représentations distribuées. [26]

IV.6.3 Réseaux non bouclés et réseaux bouclés

IV.6.3.1 Les réseaux non bouclés

Un réseau de neurones non bouclé, appelé aussi réseau *statique*, est un réseau dont le graphe des connexions est acyclique; il réalise une fonction algébrique non linéaire de ses entrées.

on utilise généralement, pour la modélisation de processus, un réseau comprenant un neurone de sortie linéaire ; un tel réseau réalise donc *une combinaison linéaire paramétrée de fonctions non linéaires paramétrées des variables*.

Si ces dernières sont les valeurs, décalées d'une période d'échantillonnage, d'un même signal, un tel réseau constitue un filtre non linéaire transverse à temps discret.

IV.6.3.2 Les réseaux bouclés

Un réseau de neurones bouclé, appelé aussi réseau *dynamique*, est un réseau dont le graphe des connexions peut contenir des cycles. Dans un réseau à temps discret, un retard (entier positif ou nul) est associé à chaque connexion; pour que le réseau soit causal, tout cycle du graphe des connexions doit être tel que la somme des retards associés à chacune des connexions du cycle soit non nul.

Il a été montré dans [31] que tout réseau (statique ou dynamique) peut être mis sous une forme particulière, appelée *forme canonique*, qui est une représentation d'état minimale. Elle est constituée d'un réseau non bouclé, et de connexions de retard unité ramenant les sorties de ce réseau non bouclé vers les entrées de celui-ci. La forme canonique permet de mettre clairement en évidence un ensemble minimum de variables d'état, et, de plus, son utilisation facilite la mise en oeuvre de l'apprentissage du réseau. Étant donné que, dans le cadre de ce mémoire, nous nous intéressons à la modélisation dynamique de processus, nous utiliserons, le plus souvent des réseaux bouclés.

IV.6.4. Réseaux non bouclés complètement connectés et réseaux à couches.

On distingue deux familles de réseaux non bouclés en fonction de la topologie des connexions entre les neurones.

IV.6.4.1 Les réseaux non bouclés complètement connectés.

Dans un réseau complètement connecté, chaque neurone reçoit les entrées du réseau et les sorties des neurones de numéro inférieur. La figure (VI.4) illustre un réseau complètement connecté ayant N_i entrées, N_c neurones cachés et une sortie.

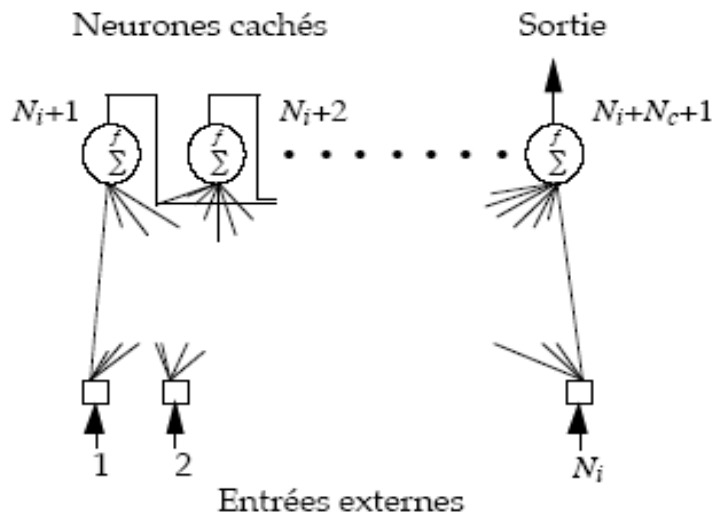


Fig. IV.4 Réseau de neurone complètement connecté

IV.6.4.2 Les réseaux non bouclés à couches.

C'est sans doute l'architecture de réseau de neurones la plus répandue. De tels réseaux sont appelés perceptrons multi-couches (ou MLP pour Multi-Layer Perceptrons). Les neurones cachés sont organisés en une (ou parfois plusieurs) couches. Ils reçoivent leurs entrées des neurones de la couche précédente (ou des entrées du réseau s'il s'agit de la première couche de neurones

cachés) et transmettent leur sortie à ceux de la couche suivante. La figure (VI.4) illustre un réseau ayant N_i entrées, constitué d'une couche de N_c neurones cachés et d'une sortie.

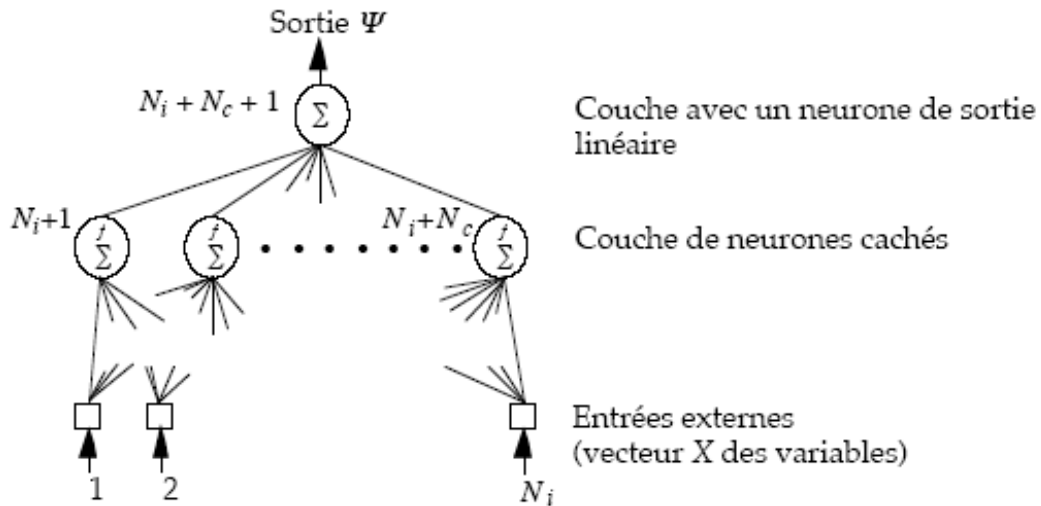


Fig. IV.5 réseau de neurone avec une couche de neurone cachés et un neurone de sortie linéaire

IV.7 Processus d'apprentissage

Parmi les propriétés désirables pour un réseau de neurones, la plus fondamentale est sûrement la capacité d'apprendre de son environnement, d'améliorer sa performance à travers un processus d'apprentissage. Mais qu'est-ce donc que l'apprentissage ? Malheureusement, il n'existe pas de définition générale, universellement acceptée, car ce concept touche à trop de notions distinctes qui dépendent du point de vue que l'on adopte.

Dans le contexte des réseaux de neurones artificiels, nous adopterons un point de vue pragmatique en proposant la définition suivante :

L'apprentissage est un processus dynamique et itératif permettant de modifier les Paramètres d'un réseau en réaction avec les stimuli qu'il reçoit de son environnement. Le type d'apprentissage est déterminé par la manière dont les changements de paramètre surviennent.

Cette définition implique qu'un réseau se doit d'être stimulé par un environnement, qu'il subisse des changements en réaction avec cette stimulation, et que ceux-ci provoquent dans le futur une Réponse nouvelle vis-à-vis de l'environnement. Ainsi, le réseau peut s'améliorer avec le temps.

Dans la plupart des architectures que nous étudierons, l'apprentissage se traduit par une modification de l'efficacité synaptique, c'est-à-dire par un changement dans la valeur des poids qui relient les neurones d'une couche à l'autre. Soit le poids $w_{i,j}$ reliant le neurone i à son entrée j . Au temps t , un changement $\Delta w_{i,j}(t)$ de poids peut s'exprimer simplement de la façon suivante :

$$\Delta w_{i,j}(t) = w_{i,j}(t+1) - w_{i,j}(t) \quad (\text{IV.2})$$

et, par conséquent, $w_{i,j}(t+1) = w_{i,j}(t) + \Delta w_{i,j}(t)$, avec $w_{i,j}(t+1)$ et $w_{i,j}(t)$ représentant respectivement les nouvelle et ancienne valeurs du poids $w_{i,j}$.

Un ensemble de règles bien définies permettant de réaliser un tel processus d'adaptation des poids constitue ce qu'on appelle l'algorithme d'apprentissage du réseau.

Il existe trois types d'apprentissages principaux. Ce sont l'apprentissage supervisé, l'apprentissage non-supervisé et l'apprentissage par tentative (graded training en anglais) [32].

On parle d'apprentissage supervisé quand le réseau est alimenté avec la bonne réponse pour les exemples d'entrées donnés. Le réseau a alors comme but d'approximer ces exemples aussi bien que possible et de développer à la fois la bonne représentation mathématique qui lui permet de généraliser ces exemples pour ensuite traiter des nouvelles situations (qui n'étaient pas présentes dans les exemples).

Dans le cas de l'apprentissage non-supervisé le réseau décide lui-même quelles sont les bonnes sorties. Cette décision est guidée par un but interne au réseau qui exprime une configuration idéale à atteindre par rapport aux exemples introduits.

Les cartes auto-organisatrices de Kohonen sont un exemple de ce type de réseau [32].

"Graded learning" est un apprentissage de type essai-erreur où le réseau donne une solution et est seulement alimenté avec une information indiquant si la réponse était correcte ou si elle était au moins meilleure que la dernière fois.

Il existe plusieurs règles d'apprentissage pour chaque type d'apprentissage. L'apprentissage supervisé est le type d'apprentissage le plus utilisé. Pour ce type d'apprentissage la règle la plus

utilisée est celle de Widrow-Hoff. D'autres règles d'apprentissage sont par exemple la règle de Hebb, la règle du perceptron, la règle de Grossberg etc. [34],[30], [32].

IV.7.1 L'apprentissage de Widrow-Hoff

La règle d'apprentissage de Widrow-Hoff est une règle qui permet d'ajuster les poids d'un réseau de neurones pour diminuer à chaque étape l'erreur commise par le réseau de neurones (à condition que le facteur d'apprentissage soit bien choisi). Un poids est modifié en utilisant la formule suivante [26]:

$$w_{k+1} = w_k + \alpha \delta_k x_k$$

- w_k est le poids à l'instant k ,
 - w_{k+1} le poids à l'instant $k+1$,
 - α est le facteur d'apprentissage,
 - δ_k caractérise la différence entre la sortie attendue et la sortie effective du neurone à l'instant k .
- x_k la valeur de l'entrée avec laquelle le poids w est associé à l'instant k .

Ainsi, si δ_k et x_k sont positifs tous les deux, alors le poids doit être augmenté. La grandeur du changement dépend avant tout de la grandeur de δ_k mais aussi de celle de x_k . Le coefficient α sert à diminuer les changements pour éviter qu'ils deviennent trop grands, ce qui peut entraîner des oscillations du poids.

Deux versions améliorées de cet apprentissage existent, la version "par lots" et la version "par inertie" (momentum en anglais) [32], dont l'une utilise plusieurs exemples pour calculer la moyenne des changements requis avant de modifier le poids et l'autre empêche que le changement du poids au moment k ne devienne beaucoup plus grand qu'au moment $k-1$.

IV.8 Les différents types de réseaux de neurones

Plusieurs types de réseaux de neurones ont été développés qui ont des domaines d'application souvent très variés. Notamment trois types de réseaux sont bien connus:

- Le réseau de Hopfield (et sa version incluant l'apprentissage, la machine de Boltzmann) ;
- Les cartes auto-organisatrices de Kohonen ;
- Les réseaux multicouches de type rétro propagation.

Le réseau de Hopfield [35], est un réseau avec des sorties binaires où tous les neurones sont interconnectés avec des poids symétriques, c'est à dire que le poids du neurone N_i au neurone N_j est égal au poids du neurone N_j au neurone N_i . Les poids sont donnés par l'utilisateur. Une application typique de ce type de réseau est le problème du voyageur de commerce, où les poids représentent d'une part les distances entre les villes et d'autre part les contraintes qui assurent la visite de chaque ville une fois et une seule. Les poids et les états des neurones permettent de définir l'"énergie" du réseau. C'est cette énergie que le réseau tente de minimiser pour trouver une solution. La machine de Boltzmann est en principe un réseau de Hopfield, mais qui permet l'apprentissage grâce à la minimisation de cette énergie. Les cartes auto-organisatrices de Kohonen [33] sont utilisées pour faire des classifications automatiques des vecteurs d'entrée. Une application typique pour ce type de réseau de neurones est la reconnaissance de parole. Les réseaux multicouches de type rétro propagation sont les réseaux les plus puissants des réseaux de neurones qui utilisent l'apprentissage supervisé.

IV.9 Le réseau de rétro propagation

Un réseau de type rétro propagation se compose d'une couche d'entrée, une couche de sortie et zéro ou plusieurs couches cachées [30]. Les connections sont permises seulement d'une couche inférieure (plus proche de la couche d'entrée) vers une couche supérieure (plus proche de la couche de sortie). Il est aussi interdit d'avoir des connections entre des neurones de la même couche.

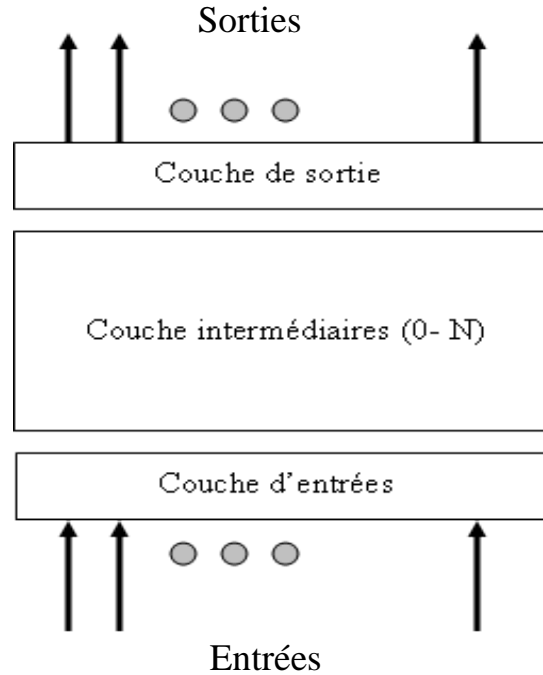


Fig. IV.6. Réseau de rétro propagation.

La couche d'entrée sert à distribuer les valeurs d'entrée aux neurones des couches supérieures, éventuellement multipliées ou modifiées d'une façon ou d'une autre.

La couche de sortie se compose normalement des neurones linéaires qui calculent seulement une somme pondérée de toutes ses entrées.

Les couches cachées contiennent des neurones avec des fonctions d'activation non-linéaires, normalement la fonction sigmoïde.

Il a été prouvé [32] qu'il existe toujours un réseau de neurones de ce type avec trois couches seulement (couche d'entrée, couche de sortie et une couche cachée) qui peut approximer une fonction $f : [0,1]^n \Rightarrow \mathbb{R}^m$ avec n'importe quelle précision $\varepsilon > 0$ désirée. Un problème consiste à trouver combien de neurones cachés sont nécessaires pour obtenir cette précision. Un autre problème est de s'assurer a priori qu'il est possible d'apprendre cette fonction.

Initialement tous les poids peuvent avoir des valeurs aléatoires, qui sont normalement très petites avant de commencer l'apprentissage. La procédure d'apprentissage se décompose en deux étapes. Pour commencer, les valeurs d'entrées sont fournies à la couche d'entrée. Le réseau propage ensuite les valeurs jusqu'à la couche de sortie et donne ainsi la réponse du réseau. A la deuxième étape les bonnes sorties correspondantes sont présentées aux neurones de la couche de sortie, qui calculent l'écart, modifient leurs poids et rétro propagent l'erreur jusqu'à la couche d'entrée pour permettre aux neurones cachés de modifier leurs poids de la même façon. Le principe de modification des poids est normalement l'apprentissage de Widrow-Hoff.

Un réseau sans couches cachées (un réseau de type ADALINE [30]) réalise une régression linéaire.

IV. 10 Régulation de la vitesse par des réseaux de neurones

Le régulateur de vitesse de la figure (II.5) est un régulateur à base de réseau de neurones. En effet, nous nous proposons dans ce qui suit de construire un réseau de neurones, qui, à chaque instant k et en fonction des deux entrées $e(k)$ et $\Delta e(k)$ fait correspondre une sortie $C_{em}(k)$. Les deux variables d'entrée de ce réseau sont normalisées entre -1 et 1. En effet, elle sont définies comme suit [17] :

$$e(k) = \frac{w_{ref}(k) - w(k)}{w_{ref}(k)}$$

$$\Delta e(k) = e(k) - e(k-1)$$

La variable de sortie $C_{em}(k)$ sera le couple de référence pour la boucle de régulation du couple.

IV.10.1 Structure du réseau de neurone

Pour notre application nous utilisons une structure à une couche (figure VI.1). Le nombre de couches de neurones cachés est donc égale à 1. En effet, cette structure à la propriété d'approximation universelle, cette couche cachée comporte 5 neurones sigmoïdes, dont les fonctions de transfert sont de type tangent hyperbolique. La sortie d'un neurone sigmoïde étant bornée entre -1 et 1, il est nécessaire, dans la deuxième couche (couche de sortie) nous utilisons un neurone dont la fonction de transfert est linéaire.

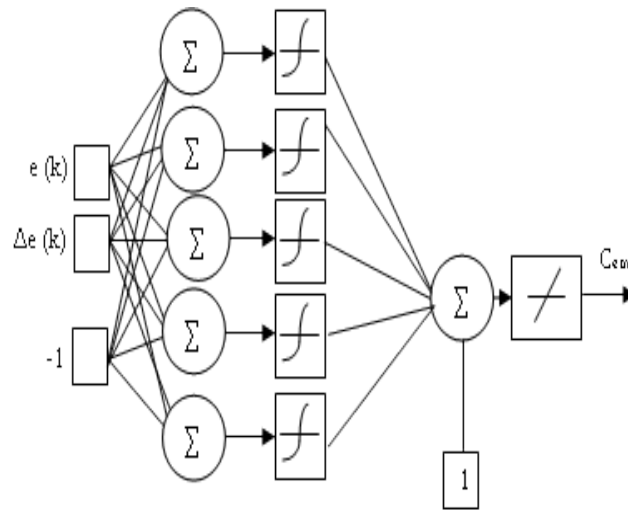


Fig. IV.7 structure de réseau neuronal utilisé pour la régulation de vitesse

IV.11 Simulation et interprétation

La simulation des comportements de la vitesse et du couple est représentée par les figures (IV.8, IV.9, IV.10). La vitesse est plus rapide et se stabilise à sa valeur de consigne. Le couple électromagnétique résultant décroît progressivement ou il prendra la valeur nulle dès que la vitesse se stabilise. La figure (IV.11), montre la comparaison des vitesses de différents neurones dans la couche cachée. Il est clair que plus augmente le nombre de neurone dans la couche cachée, plus que le régime transitoire de la vitesse est plus rapide.

Les résultats de simulation des figures (IV.12, IV.13, IV.14, IV.15, IV.16) illustrent l'évolution de la vitesse de rotation, du couple, des flux rotoriques et des courants de phase en présence et absence des variations des paramètres de système. On présente d'abord un démarrage optimal présenté par la figure (IV.12), ensuite application d'une charge à $t=0.5$ s (IV.13), la vitesse atteint sa valeur de référence, les flux gardent leurs orientations et le couple électromagnétique suit le couple de charge et de référence. Une simulation d'un démarrage lors de l'augmentation du moment d'inertie de trois fois de la valeur nominale a été faite (figure IV.15). La vitesse se stabilise en un temps plus court. Enfin les figures (IV.16), montrent les

effets de l'augmentation de la résistance rotorique de 50% de la valeur nominale. L'influence de ce type de variation est remarquable dans les flux et le couple.

IV.12 Conclusion

La technique de réglage par réseau de neurone, permet à la commande d'être insensible à de telles variations, la vitesse est peu affectée par les perturbations et suit sa consigne. On peut conclure que la commande par réseau de neurone associée à la commande vectorielle se prête bien pour le réglage de la vitesse de la machine asynchrone. Elle assure une bonne orientation du flux rotorique et donc un bon découplage entre le flux et le couple.

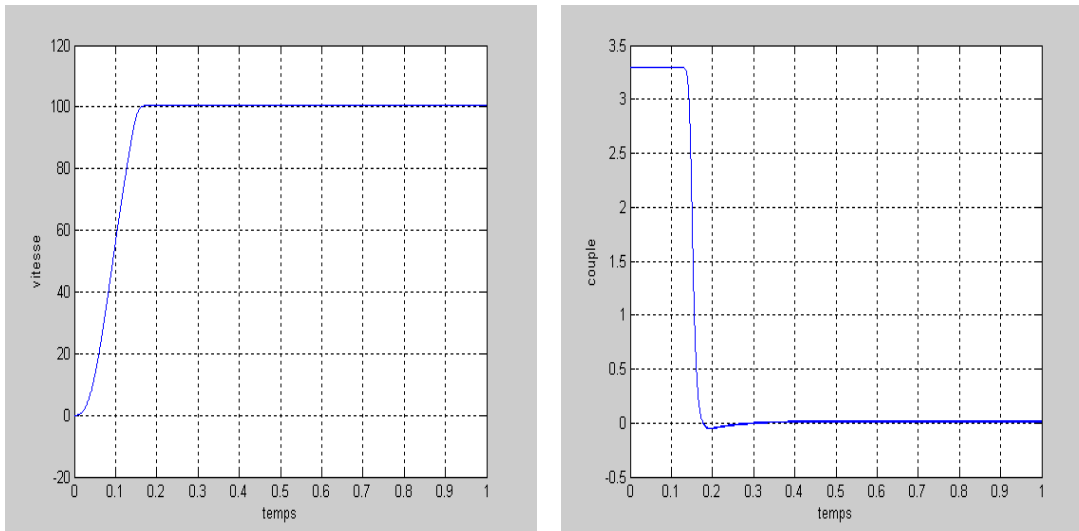


Fig.IV.8 vitesse et couple de 3 neurones

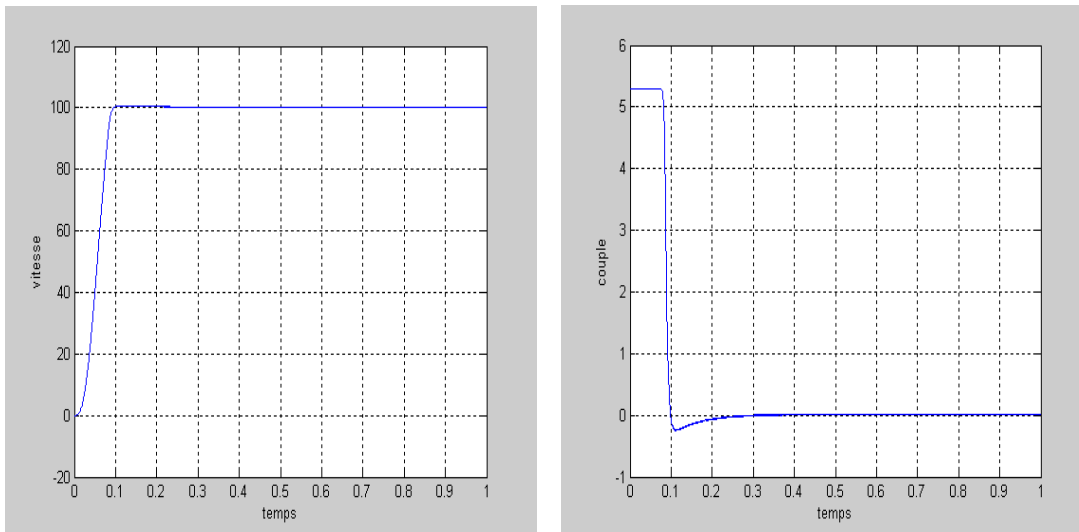


Fig. IV.9 Vitesse et couple de 5 neurones

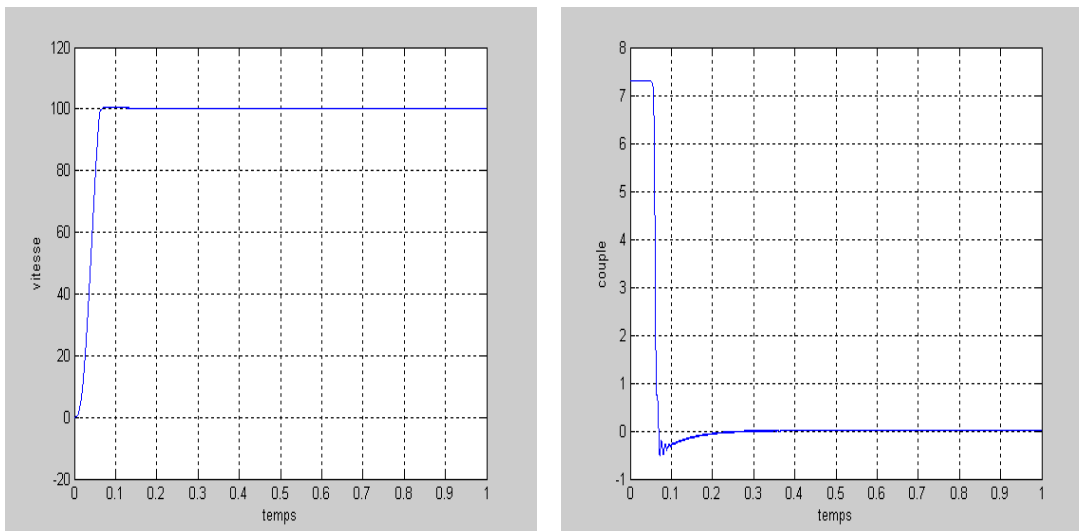


Fig. IV.10 Vitesse et couple de 7 neurones

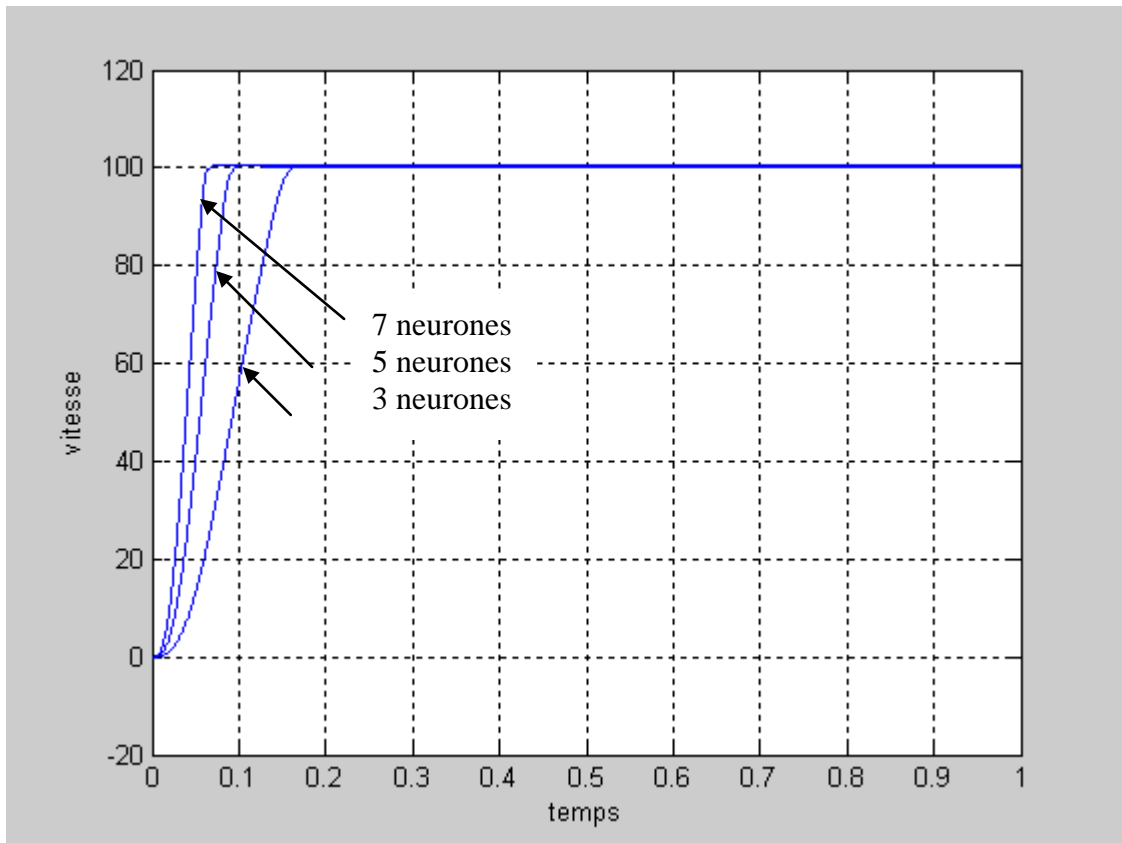


Fig. IV.11 Comparaison des vitesses de régulateur neurone

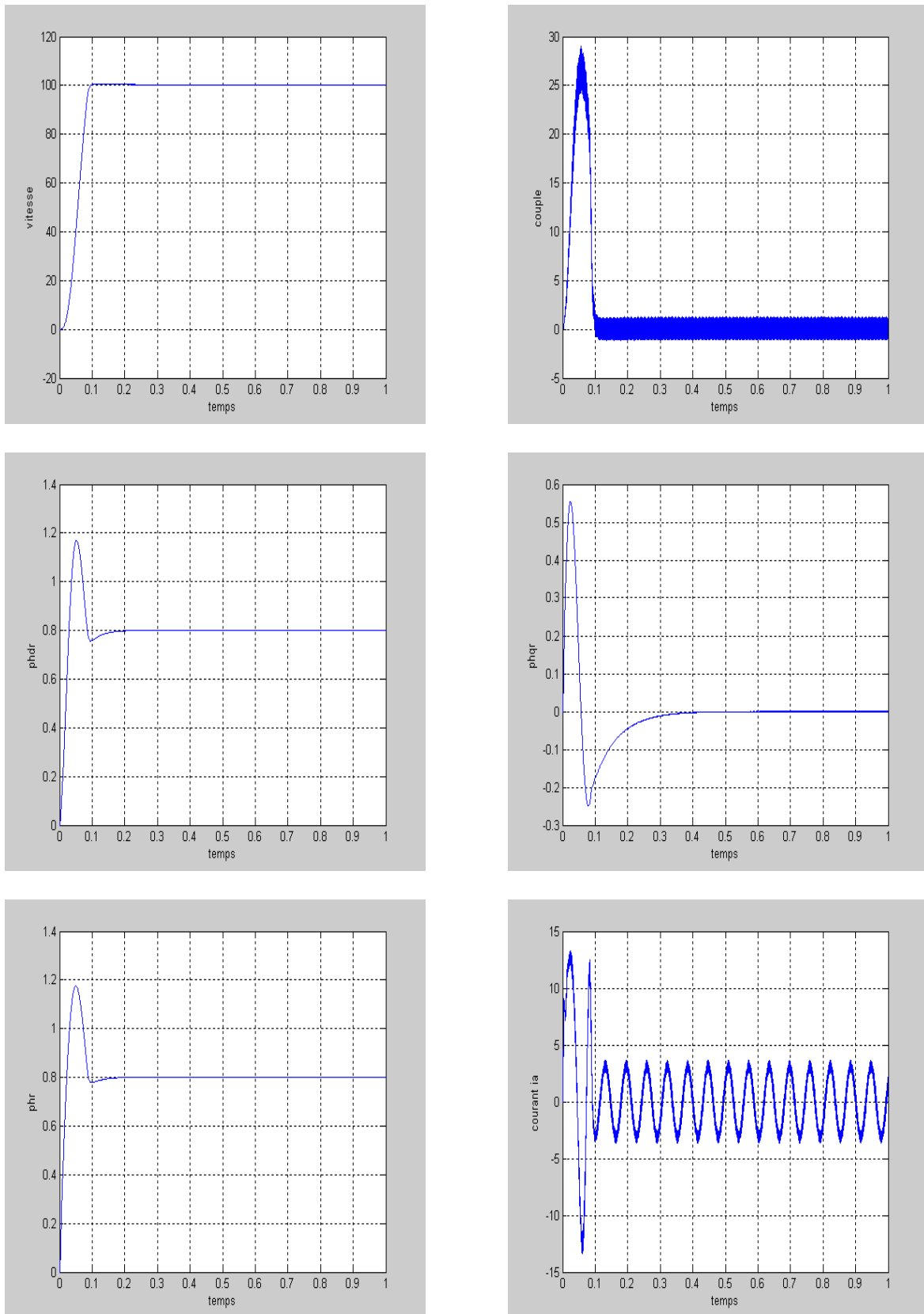


Fig.IV.12 Réglage par réseau de neurone de la vitesse lors d'un démarrage à une Vitesse de 100 rd/s

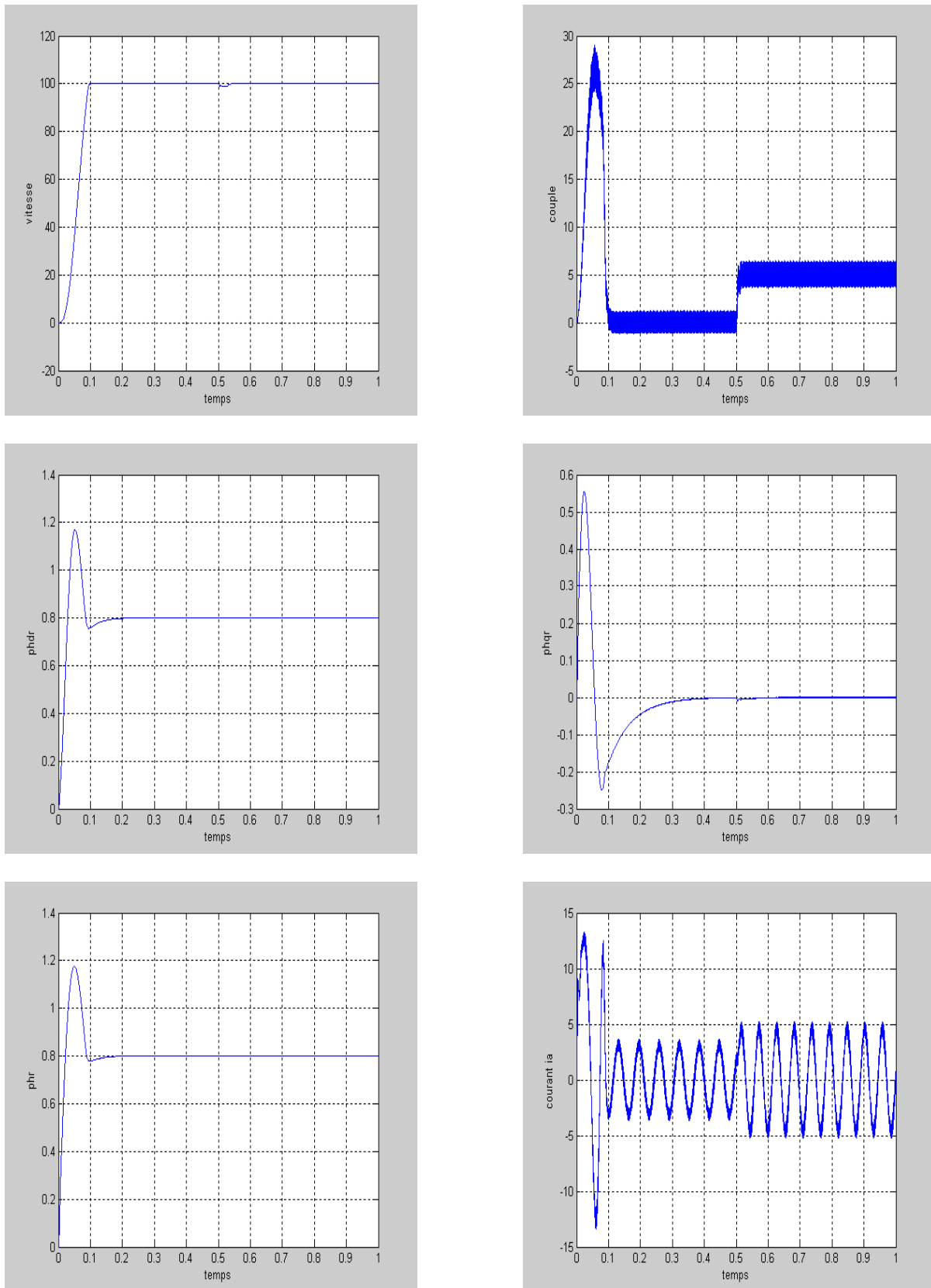


Fig.IV.13 Réglage par réseau de neurone de la vitesse avec application d'un couple de charge de (5 N.m) à $t=0.5$ s

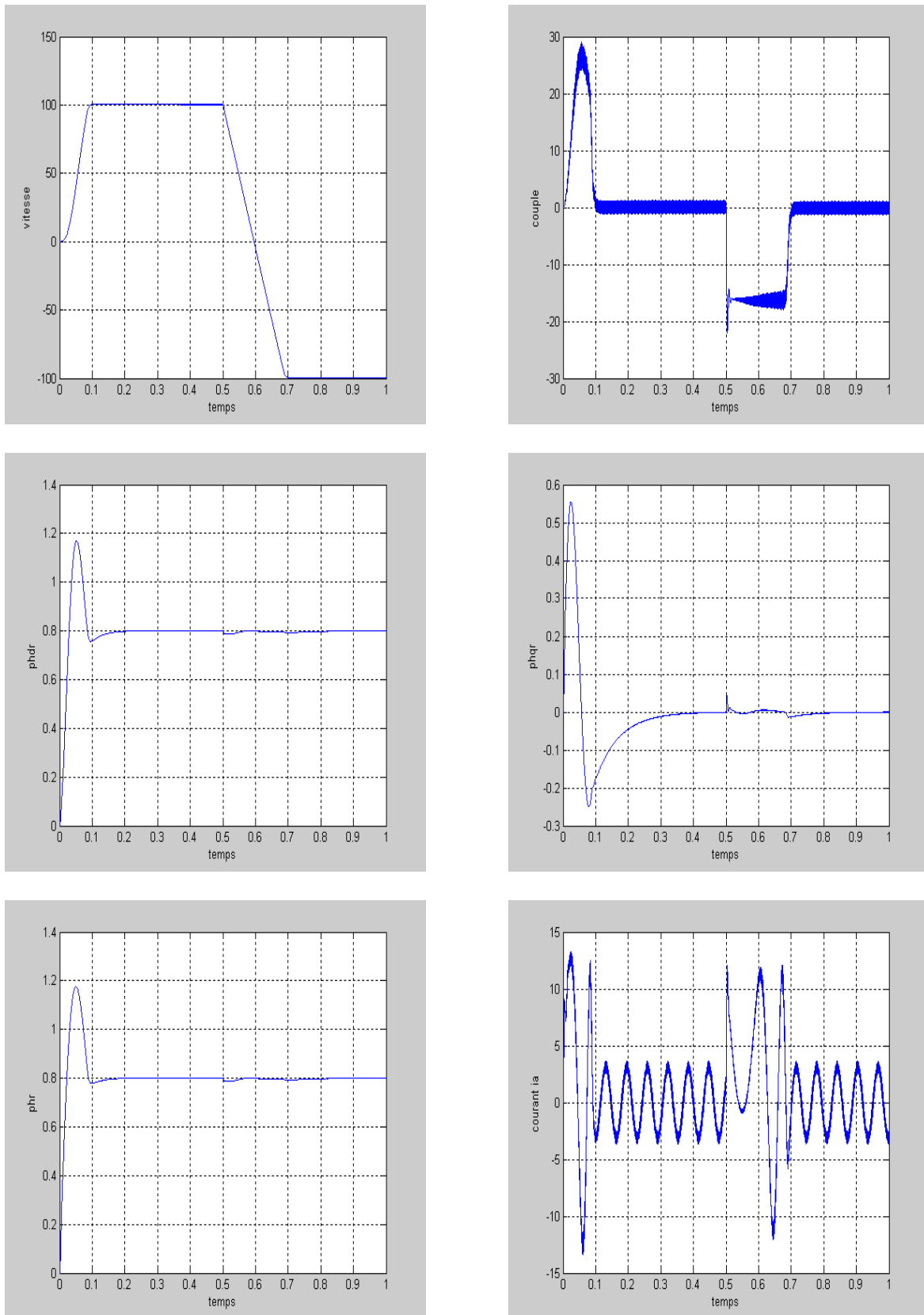


Fig. IV.14 Réglage par réseau de neurone de la vitesse lors d'un démarrage à une Vitesse de 100 rd/s et inversement de sens de rotation à $t=0.5$ s

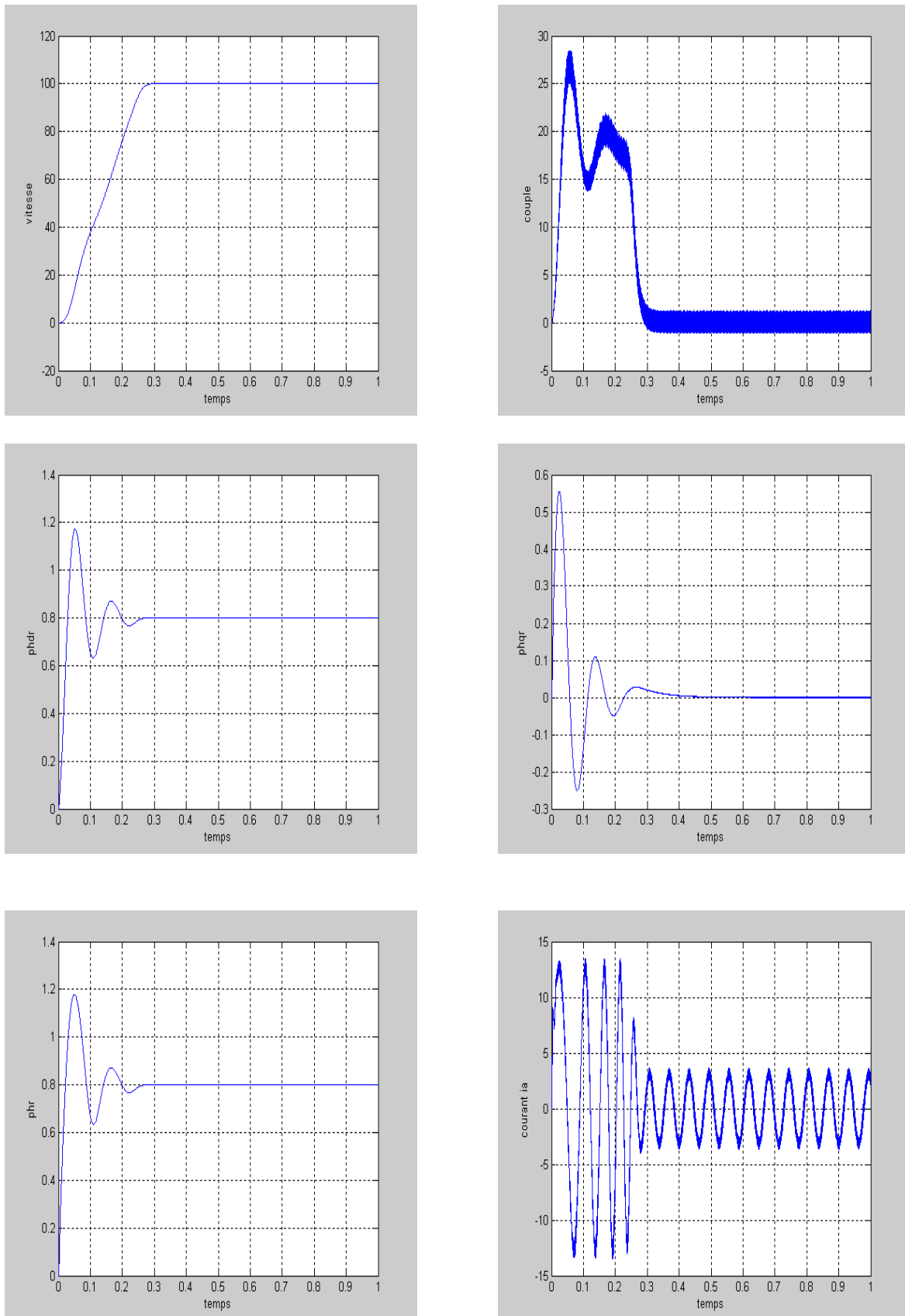


Fig. IV.15 Réglage par réseau de neurone de la vitesse avec augmentation de moment d'inertie de $3 \cdot J_n$

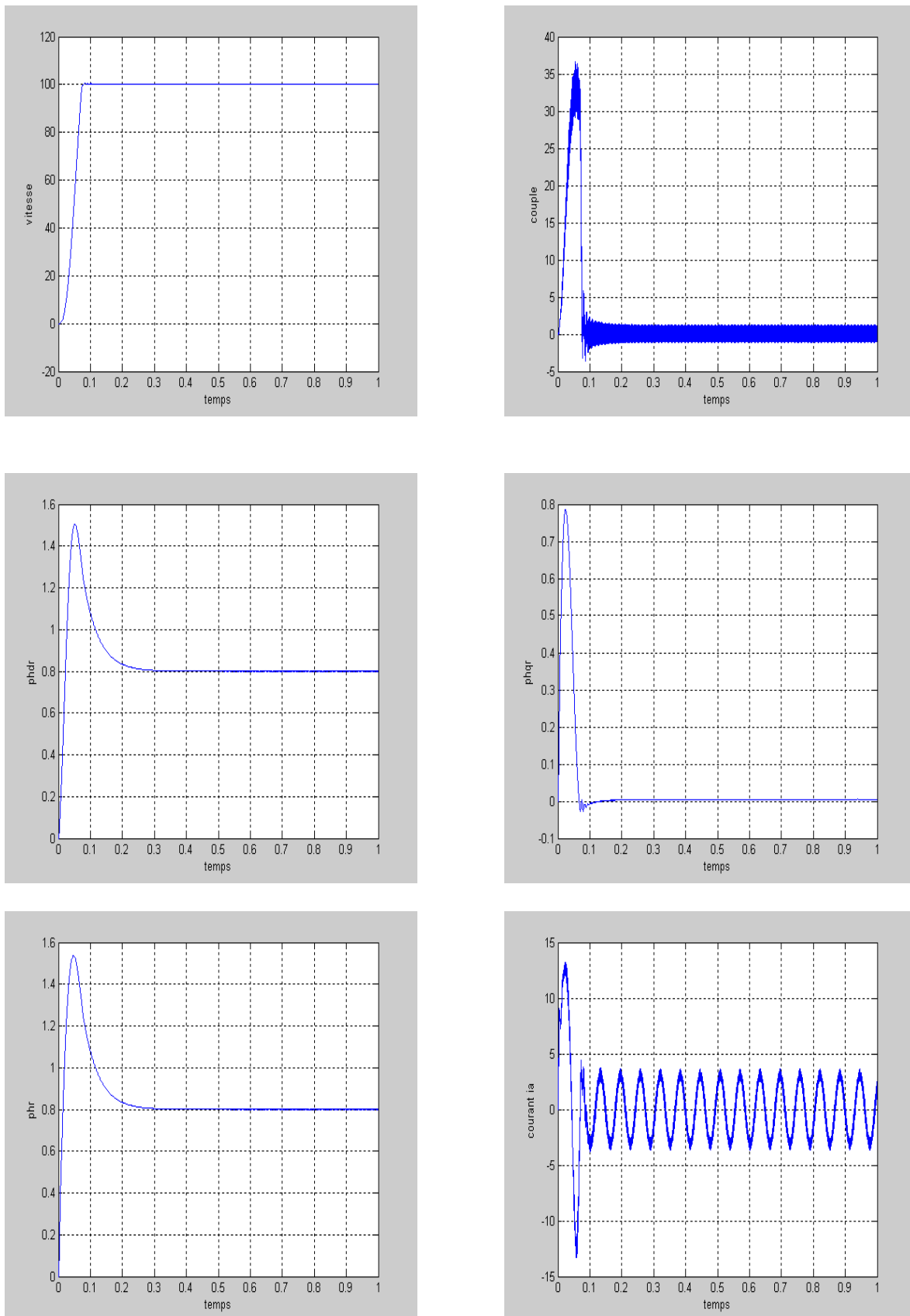


Fig. IV.16 Réglage par réseau de neurone de la vitesse avec augmentation de la résistance rotorique de 50% R_{rn}

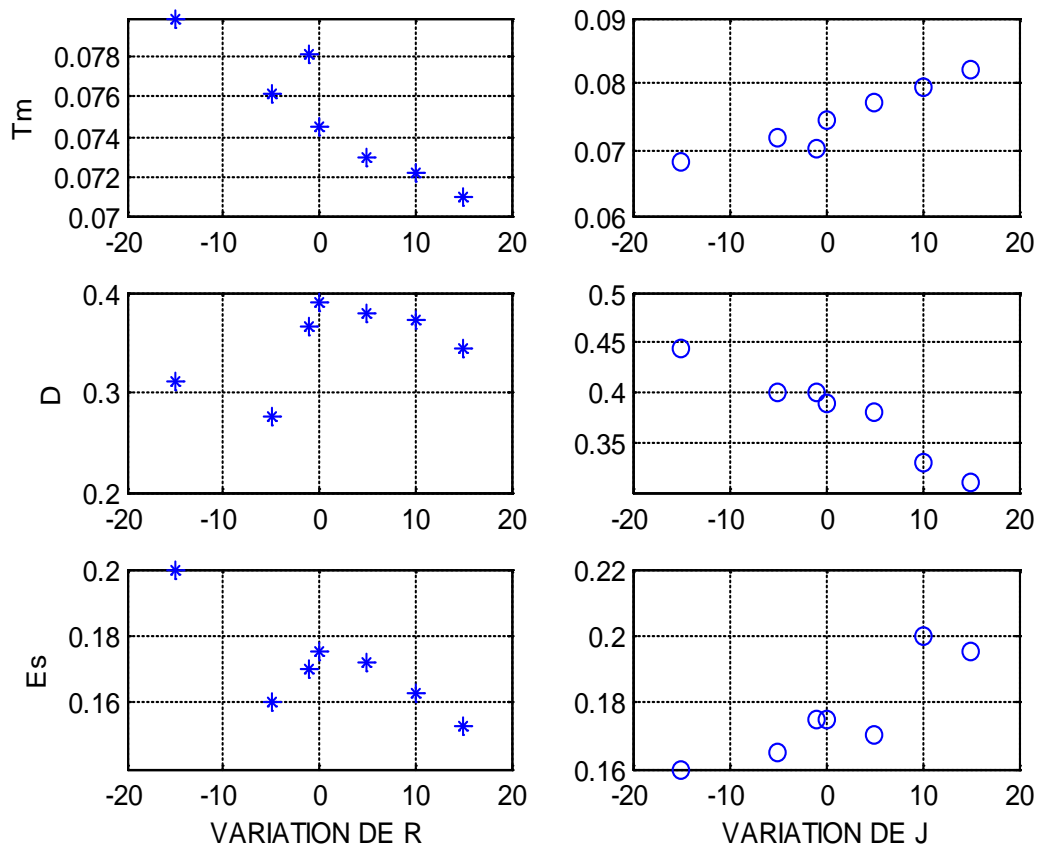
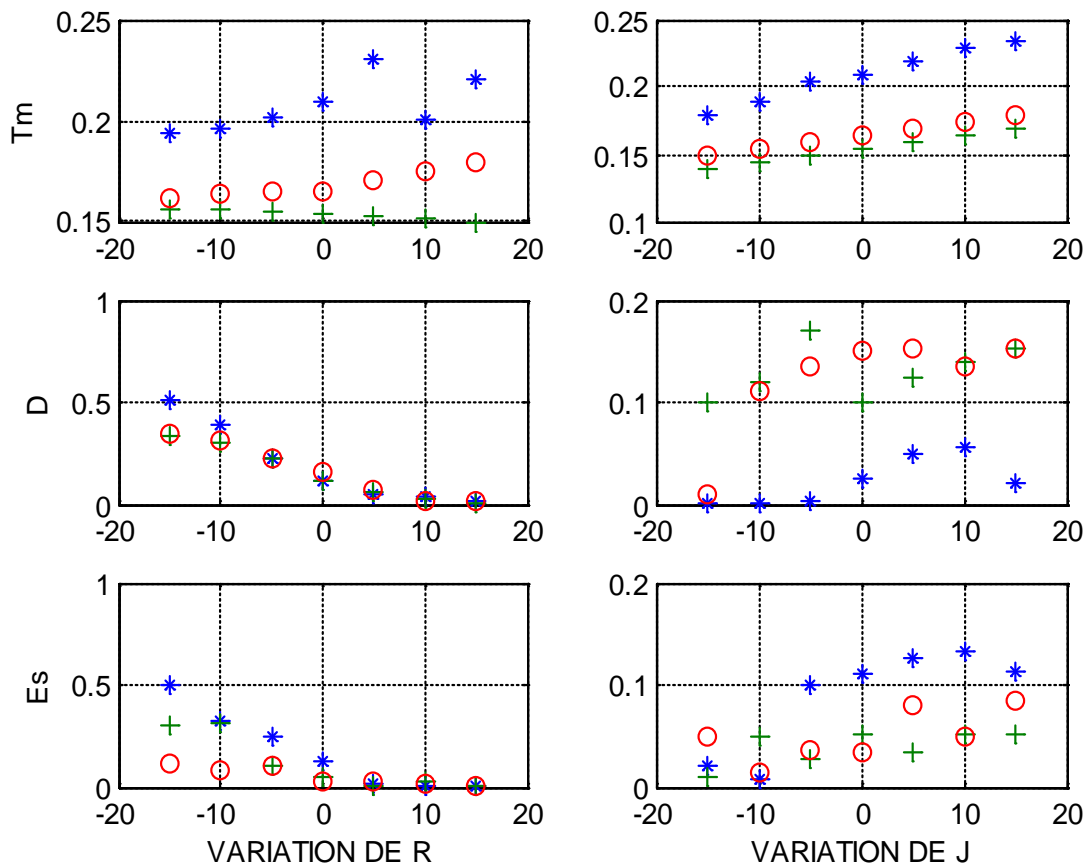


Figure IV.17 Comparaison de T_m , D , E_s de régulateur PI

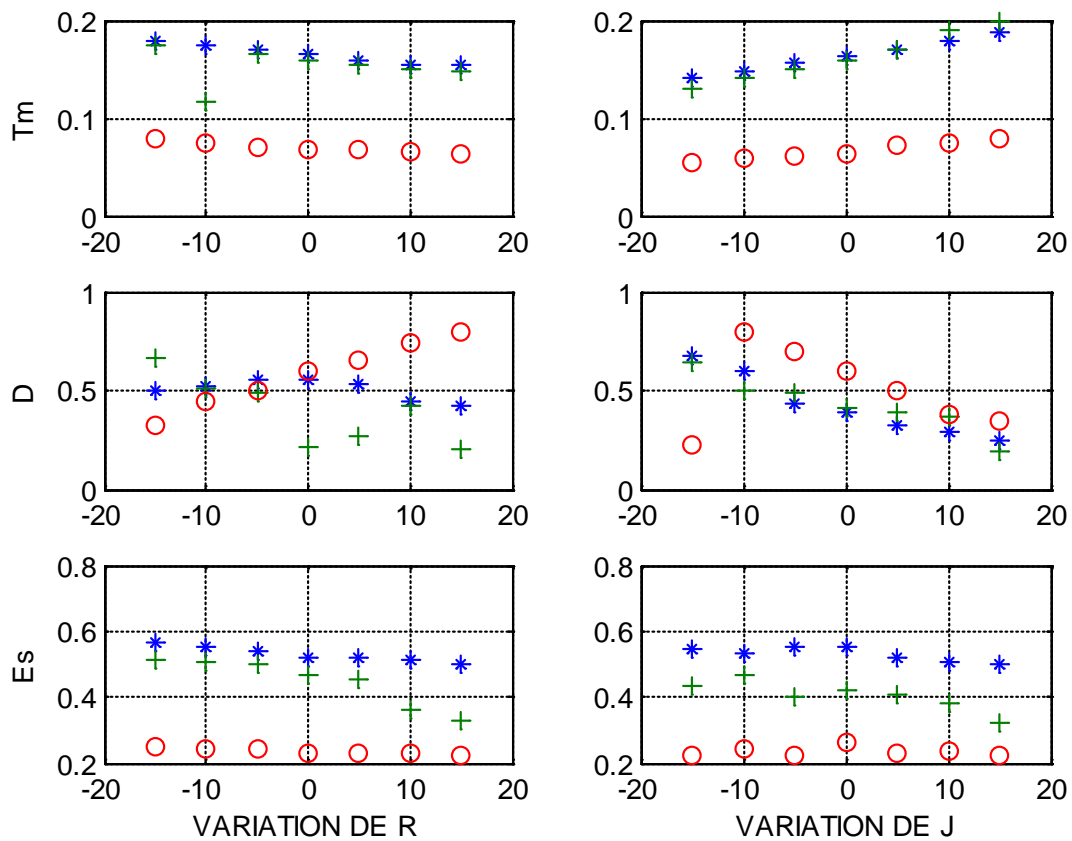
Le temps de montée est inversement proportionnel avec l'augmentation de R , le dépassement et l'erreur statique répendent approximation à la même loi relativement à la variation de R . plus le moment d'inertie augmente plus le système devient long. La variation de T_m est proportionnel à celle de J , concernant le dépassement sa valeur est inversement proportionnelle à la variation de J et le système devient plus amortie pour des J important. Finalement, l'erreur statique est importante avec l'augmentation de J



‘*’ Trois ensembles flous, ‘+’ Cinq ensembles flous, ‘o’ Sept ensembles flous

Figure IV.18 Comparaison de T_m , D , E_s de régulateur floue

D’abord, on remarque que l’augmentation de R et celle de J provoque un retard du système. Mais, il est intéressant de remarquer que la variation de moment d’inertie ne présente aucune sensibilité aussi bien que, le dépassement reste insensible à la variation de R alors pour les variations de J, le dépassement augmente relativement à la valeur nominale. Finalement l’erreur statique reste insensible. Pour des augmentations positives et légèrement importantes pour la variation de J



‘*’ Trois neurones, ‘+’ Cinq neurones, ‘o’ Sept neurones

Figure IV.19 Comparaison de T_m , D , E_s du régulateur neurone

On remarque qu’avec l’augmentation de la résistance R et le moment d’inertie le temps de montée pour (3 neurones) et (5 neurones) sensiblement identique. Concernant, le dépassement sa sensibilité et légèrement dépend de la variation de R et J. alors que l’erreur statique n’a pas de changement notable.

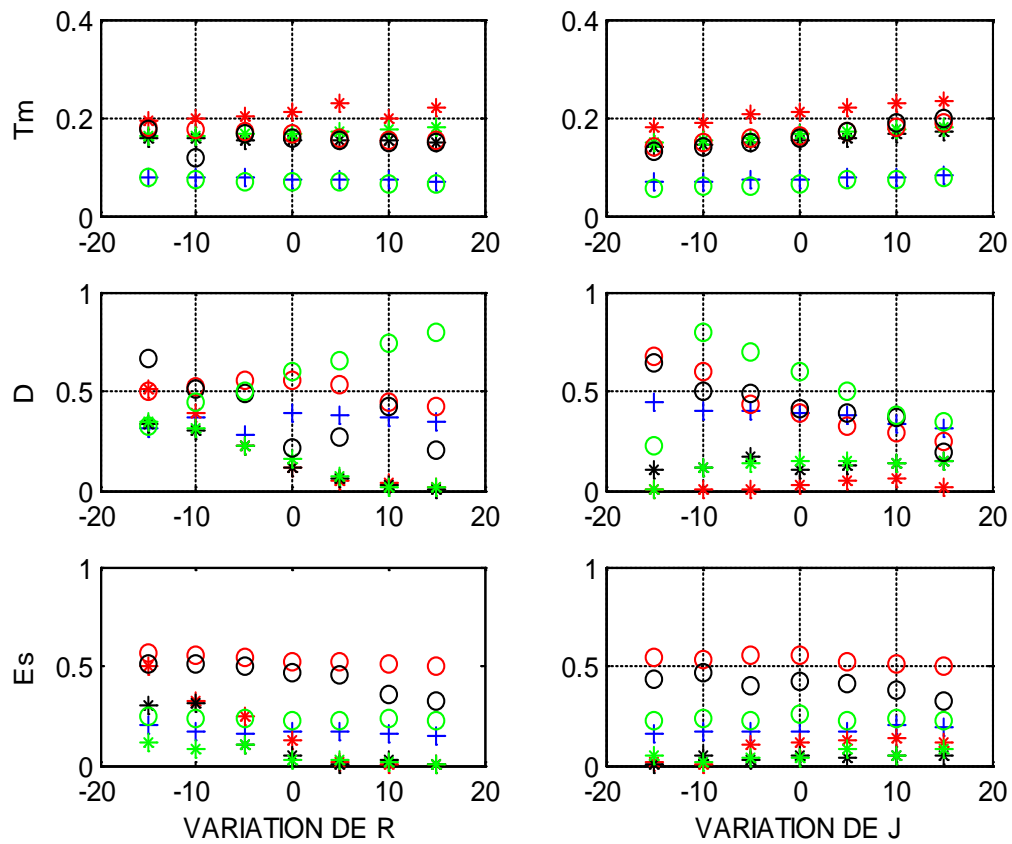


Figure IV.20 Comparaison de T_m , D , E_s des trois régulateurs

- ‘*’ Trois ensembles flous.
- ‘*’ Cinq ensembles flous.
- ‘*’ Sept ensembles flous.
- ‘+’ régulateur PI
- ‘o’ Trois neurones.
- ‘o’ Cinq neurones.
- ‘o’ Sept neurones.

Conclusion générale

Le travail présenté dans ce mémoire constitue une contribution à la mise en œuvre de techniques non conventionnelles de commande numérique. Nous avons ainsi élaboré deux stratégies de contrôle de vitesse des machines asynchrones basées sur la logique floue et les réseaux de neurones. Les résultats obtenus nous permettent de montrer que ces techniques, jusqu'à présent peu utilisées, surtout en milieu industriel, sont bien adaptées lorsqu'on a des contraintes sévères au niveau de la régulation. Les systèmes de contrôle de vitesse des machines asynchrones utilisant la logique floue ou les réseaux de neurones présentent des performances intéressantes.

Listes de figures

Figures	Titres	N
Fig.I.1	Représentation des enroulements statorique et rotorique	5
Fig.I.2	Simulation du démarrage sous pleine tension à vide	12
Fig.I.3	Démarrage sous pleine tension avec application d'un couple Résistant ($c_r = 5 \text{ N.m}$) à l'instant $t=1\text{s}$	13
Fig.I.4	Schéma d'un onduleur de tension alimentant le stator du MAS	16
Fig.I.5	MLI sinus –triangle bipolaire	18
Fig.I.6	Signaux de commande et de sortie de l'onduleur	19
Fig.I.7	Association convertisseur – machine à vide	21
Fig.I.8	Association convertisseur – machine avec une charge de 5 Nm à l'instant $t=1 \text{ s}$	22
Fig.I.9	Association convertisseur – machine avec application d'une charge à $t=0.5\text{s}$ et $t=0.8\text{s}$ et inversement de sens de rotation	23
Fig.II.1	Repère d q lié au flux rotorique	25
Fig.II.2	Structure du bloc F.O.C	28
Fig.II.3	Reconstitution des tensions v_{ds} et v_{qs}	30
Fig.II.4	Commande découplée – expression de ϕ_r et c_e	30
Fig.II.5	Schéma bloc de la commande vectorielle indirecte	31
Fig.II.6	Schéma fonctionnel de régulation de courant i_{ds} .	32
Fig.II.7	Schéma fonctionnel de régulation de vitesse	34
Fig.II.8	Commande vectoriel indirect, application d'une charge à $t=1.5\text{s}$ et $t=4.5\text{s}$ avec une vitesse de 100 rd/s	35
Fig.II.9	Commande vectorielle directe de flux d'une machine alimentée en tension	38
Fig.II.10	Schéma en boucle ouverte	39
Fig.II.11	Schéma en boucle ouverte	40
Fig.II.12	Schéma en boucle fermée	41
Fig.II.13	Commande vectoriel direct, application d'une charge à $t=1\text{s}$ et $t=2 \text{ s}$	42

Fig.III.1	Schéma générale d'un bloc fonctionnel flou	46
Fig.III.2	Niveaux d'appartenance en logique booléenne	46
Fig.III.3	Niveaux d'appartenance en logique floue	47
Fig.III.4	Différente formes des fonction d'appartenance	48
Fig.III.5	Représentation des variables linguistiques	50
Fig.III.6	Univers de discours normalisé dans l'intervalle $[-1 \ +1]$ avec une partition de 3, 5,7 ensembles flous	52
Fig.III.7	Exemple d'inférence à plusieurs règles	53
Fig.III.8	Structure d'un régulateur à logique floue (RLF)	56
Fig.III.9	Description des fonctions d'appartenance	58
Fig.III.10	Vitesse et couple de 3 sous ensembles	62
Fig.III.11	Vitesse et couple de 5 sous ensembles	62
Fig.III.12	Vitesse et couple de 7 sous ensembles	62
Fig.III.13	Comparaison des sous ensembles 3,5 et 7	63
Fig.III.14	Réglage par logique floue de la vitesse lors d'un démarrage à vide pour une vitesse de 100 rd/s	64
Fig.III.15	Réglage par logique floue de la vitesse avec application d'un couple de charge à $t= 5$ s	65
Fig.III.16	Réglage par logique floue de la vitesse avec inversion de sens de rotation à $t=0.5$ s	66
Fig.III.17	Réglage par logique floue de la vitesse avec augmentation du moment d'inertie de $3 \cdot J_n$	67
Fig.III.18	Réglage par logique floue de la vitesse avec augmentation de la résistance rotorique de $50\% R_{rn}$	68
Fig.IV.1	le neurone artificiel	71
Fig.IV.2	Fonctions de transfert	72
Fig.IV.3	Topologie des réseaux de neurone	74
Fig.IV.4	Réseau de neurone complètement connecté	78
Fig.IV.5	Réseau de neurone avec une couche de neurone cachés et un neurone de sortie linéaire	79
Fig.IV.6	Réseau de rétro propagation.	83
Fig.IV.7	Structure de réseau neuronal utilisé pour la régulation de vitesse	85
Fig.IV.8	vitesse et couple de 3 neurones	87

Fig.IV.9	Vitesse et couple de 5neurones	87
Fig.IV.10	Vitesse et couple de 7neurones	87
Fig.IV.11	Comparaison des vitesses de régulateur neurone	88
Fig.IV.12	Réglage par réseau de neurone de la vitesse lors d'un démarrage à une Vitesse de 100 rd/s	89
Fig.IV.13	Réglage par réseau de neurone de la vitesse avec application d'un couple de charge de 5 n.m à $t=0.5$ s	90
Fig.IV.14	Réglage par réseau de neurone de la vitesse lors d'un démarrage à une Vitesse de 100 rd/s et inversement de sens de rotation à $t=0.5$ s	91
Fig.IV.15	Réglage par réseau de neurone de la vitesse avec augmentation de moment d'inertie de $3*J_n$	92
Fig.IV.16	Réglage par réseau de neurone de la vitesse avec augmentation de la résistance rotorique de $50\%R_{rn}$	93
Fig.IV.17	Comparaison de T_m , D, Es de régulateur PI	94
Fig.IV.18	Comparaison de T_m , D, Es de régulateur floue	95
Fig.IV.19	Comparaison de T_m , D, Es du régulateur neurone	96
Fig.IV.20	Comparaison de T_m , D, Es des trois régulateurs	97

Nomenclature

Symboles	Désignations
(r)	Grandeur rotor ou de repère rotorique (R)
(s)	Grandeur stator ou de repère statorique (S)
(d)	Axe d du repère tournant (d, q)
(q)	Axe q du repère tournant (d, q)
p	Nombre de paire de pôles
R _r	Résistance rotorique
R _s	Résistance statorique
M	Inductance mutuel
σ	Coefficient de dispersion
v_{ds}	Tension statorique instantanée dans l'axe d
v_{qs}	Tension statorique instantanée dans l'axe q
i_{ds}	Courant statorique instantanée dans l'axe d
i_{qs}	Courant statorique instantanée dans l'axe q
J	Moment d'inertie
Ω_r	Vitesse angulaire mécanique du rotor
k_f	Coefficient de frottement
c_e	Couple électromagnétique délivré par le moteur
c_r	Couple résistant
ω_r	Vitesse de rotation électrique
e	Erreur entre grandeur réel et de référence
Δe	Dérivé de l'erreur
S	Opérateur de Laplace
x	L'état d'activation du neurone
f	Fonction d'activation du neurone
w_n	Poids synaptique
n_h	Nombre d'unités cachées du réseau

N_i	Nombre de <i>Neurone d'entrées</i>
N_c	Nombre de Neurons cachés
w_k	Poids à l'instant k
w_{k+1}	Le poids à l'instant k+1
α	Facteur d'apprentissage
δ_k	Caractérise la différence entre la sortie attendue et la sortie effective du neurone à l'instant k.
x_k	La valeur de l'entrée avec laquelle le poids w est associé à l'instant k
μ	Fonction d'appartenance
E_1	Sous ensemble
X	Nom de variable
$T(x)$	Ensemble de terme de variable
U	Univers de discours
ΔU	Décisions de la sortie

Nomenclature

- (r) Grandeur rotor ou de repère rotorique (R)
- (s) Grandeur stator ou de repère statorique (S)
- (d) Axe d du repère tournant (d, q)
- (q) Axe q du repère tournant (d, q)
- p nombre de paire de pôles
- R_r résistance rotorique
- R_s résistance statorique
- M inductance mutuel
- σ Coefficient de dispersion
- v_{ds} Tension statorique instantanée dans l'axe d
- v_{qs} Tension statorique instantanée dans l'axe q
- i_{ds} Courant statorique instantanée dans l'axe d
- i_{qs} Courant statorique instantanée dans l'axe q
- J moment d'inertie
- Ω_r Vitesse angulaire mécanique du rotor
- k_f Coefficient de frottement
- c_e Couple électromagnétique délivré par le moteur
- c_r Couple résistant
- ω_r Vitesse de rotation électrique
- e Erreur entre grandeur réel et de référence
- Δe Dérivé de l'erreur
- S opérateur de Laplace
- x L'état d'activation du neurone
- f Fonction d'activation du neurone
- w_n Poids synaptique
- n_h Nombre d'unités cachées du réseau
- N_i Nombre de Neurone d'entrées
- N_c Nombre de Neurones cachés

w_k Poids à l'instant k

w_{k+1} Le poids à l'instant k+1

α Facteur d'apprentissage

δ_k Caractérise la différence entre la sortie attendue et la sortie effective du neurone à l'instant k.

x_k La valeur de l'entrée avec laquelle le poids w est associé à l'instant k

μ fonction d'appartenance

E_1 sous ensemble

X nom de variable

$T(x)$ ensemble de terme de variable

U univers de discours

ΔU décisions de la sortie

Bibliographie

- [1] JORDA X. « Conception et réalisation d'une commande économique de couple d'une machine asynchrone pour la traction électrique »,Thèse CEGELY, Lyon, INSA de Lyon, 1995.
- [2] Franck Morand « Techniques d'observation sans capteur de vitesse en vue de la commande des machines asynchrones » ,thèse de doctorat université de lion, 2005
- [3] G. Guy, C .Guy, « Actionneur électrique, principe modèles commande », Edition Eyrolles, 2000.
- [4] G. lacraux, « Actionneur électrique pour la robotiques et les asservissements », Lavoisier, 1995.
- [5] G.Seguir, F. Notelet, « Electrotechnique industrielle », technique et documentation Lavoisier 1977.
- [6] Carlos Canudas de Wit, « modélisation contrôle vectoriel et DTC », commande des moteurs asynchrones volume 1, 2001.
- [7] Jellasi K. « Positionnement d'une machine asynchrone par la méthode du flux orienté », thèse de doctorat. INPT, Toulouse, 1991.
- [8] Ourth T , Nguyen phuoc V.T, Pietrzak- David M. et de Fornel b. « Sensorless Speed Control of Induction Motor Drives Using observer Based vector control », In Proceeding of the Conference ICEM 92 Manchester, 1992.
- [9] leonhard w. « control of a. c. machine with the help of microelectronics », In 3 rd IFAC Symposium on control. Inpow. 1983.
- [10] Roboam X, Hapiot J.C, de Fornel B. et Andrieux C. « contrôle d'une machine asynchrone par estimation robuste de la vitesse », Journal de physique III.2 : 439-453,1992.
- [11] Von Westerholt E. « Commande non linéaire d'une machine asynchrone », thèse de doctorat, INPT, Toulouse, 1994.
- [12] xavier roboam, « variateur de vitesse pour machine asynchrone, contrôle de vitesse sans capteur mécanique », thèse de doctorat de l'INPT, toulouse 1991.
- [13] O. Landolt, P. Marchal et E. Vittoz, « processeurs flous », Eisi-Micro Swiss Semaine d'enseignement spécifique, Exposé on st-Inier, Décembre 2000.
- [14] J. C de Barros, « Application de la logique floue à la commande optimale du moteur asynchrone », thèse de doctorat, université Aix- Marseille III, France, 2003.
- [15] K. Benlarbi, « optimisation floue, neuronale et neuro-floue d'un système de pompage photovoltaïque par déférentes machines », thèse de magistère, université de Batna, 2003

- [16] I. Baghli , « contribution à la commande de la machine asynchrone, utilisation de la logique floue , des réseaux de neurone et des algorithmes génétique », thèse de doctorat, université Henri Poincaré, France, 1999.
- [17] BEN ABDELGHANI A. « Application des techniques intelligentes pour le contrôle en vitesse des machines asynchrone ». DEA A.T.S , ENIT, 09/99
- [18] Earl Cox, « fuzzy fundamentals », IEEE spectrum, pp 58-61 octobre 1992.
- [19] B. Mokhtari, « Implantation de la DTFC par des techniques de l'intelligence artificielle neuronale et floue : Application à la machine asynchrone alimentée par des onduleurs à deux et trois niveau », thèse de magistère, université de Batna, 2004.
- [20] R. Benabdenour, P. Borne, M. Ksouri, et F. M'sahli « Identification et Commande des procédés Industrielles », Edition technique, Paris 2001.
- [21] M.E. Haque, And M. F. Rahman, « Influence of stator resistance variation on direct torque controlled interior Permanent Magnet Synchronous Motor Drive performance an dits compensation », In Proceedings of the thirty-sixth IEEE Industry Applications Conférence (IAS- 2001) , vol. 4, pp. 2563-2569, chicago, USA, 2001.
- [22] H. Buhler, « réglage par logique floue », collection Electricité, presse polytechnique et universitaire romandes, 1994.
- [23] P.A.Arias, « Improvements in direct torque control of induction motors », thèse de doctorat, université de catalunya, Espagne, 2000
- [24] Marc Parizeau. Réseaux De Neurone, GIF-21140 et GIF-64326, Automne 2004, université Laval.
- [25] Fabien Moutard , « Introduction aux réseau de neurone et a leur application en automatique », école des mines de paris, décembre 2003.
- [26] Kary FRÄMLIN, « Les réseaux de neurones comme outils d'aide à la décision floue », Rapport de D.E.A, Ecole Nationale Supérieure des Mines de Saint-Etienne.
- [27] Introduction aux réseaux de neurones, considérations actuarielles, 5255 Av. Decelles, Montréal (Québec) H3T 2B1
- [28] Yacine OUSSAR, « Réseaux d' ondellettes et réseaux de neurones pour la modélisation statique et dynamique de processus ». THÈSE de DOCTORAT de l' UNIVERSITÉ PIERRE ET MARIE CURIE, 1998.
- [29] O. Nerrand, P. Roussel-Ragot. L. Personnaz, G. Dreyfus, *Neural Networks and Nonlinear Adaptive Filtering: Unifying Concepts and New Algorithms*, Neural Computation, 5 (2) (1993) 165-199.

- [30] Rumelhart David E, McClelland James L, et al , « Parallel Distributed Processing »,vol 1 .
Massachusetts : MIT Press 1988. 547 p.
- [31] nerrand, P,roussel , d urbain, L,Personnaz,G , DREYFUS , training recurrent network :
why and how ?an illustration in process modeling, IEEE trans, on neural network 5 (2) (1994)
178- 184.
- [32] Hecht- Nielsen robert neurocomputing. Addison- wesley publishing company, 1990.
433 p.
- [33] kohonen tervo, « Self- Organisation and Associative Memory ». Berlin : spring- verlag,
1984.
- [34] fogelnon- soulie, francois méthode connexionnistes pour l'apprentissage. Acts des
journées nationales sur l'intelligence artificielle. Paris : teknea, 1988. p. 275- 293.
- [35] Hopfield, J, J , « Neoronal Networks and Physical système with Emergent collective
Computational Abilites ». Proceedings of the National Academy of Science. USA, 1982. vol
79, p. 2554- 58.