



République Algérienne Démocratique et  
Populaire Ministère de l'Enseignement Supérieur et de la Recherche  
Scientifique

**Université Abbes Laghrour Khenchela**  
**Faculté de Technologie**  
**Département d'Electrotechnique**

## **MEMOIRE**

**Présentée pour l'obtention du diplôme de**  
**MASTER**  
**Spécialité: Electrotechnique**  
**Option: Commande**  
**Electrique**

## **Thème**

**CONTRÔLE ET OPTIMISATION DE LA REGULATION DE**  
**VITESSE D'UNE MOTEUR A COURANT CONTINU**

***Présenté par:***

- ✓ REMADNIA SOUAAD
- ✓ OUADI AMINA

***Encadreur:***

TAREK BOUTABBA

***Devant Le Jury Composé De***

- ✓ SAIDI ABEDLKADER
- ✓ BENHADDA NABIL

***2019/2020***

# *Remerciement*

*Avant tout nous remercions Dieu le tout puissant de nous avoir donnés le courage, la volonté, la patience, et la santé durant toutes ces années et que grâce à lui ce travail a pu être réalisé.*

*Nous avons l'honneur et le plaisir de témoigner notre reconnaissance à tous ceux qui ont collaboré de près ou de loin à la réalisation de ce projet, ou qui ont soutenu l'effort par des conseils et des encouragements toujours bienvenus.*

*Nos remerciements donc, en premier lieu, notre encadreur : Dr. Tarek Boutabba qui grâce à dieu et à son appui le plus cher, ce projet a vu le jour, par son appui et ses conseils.*

*Nous remercions tous les enseignants du Faculté option : Electrotechnique, Université Abbes Laghrour – Khenchela*

*Sans oublier Nos Parents, Nos frères, Nos sœurs,  
et Tous les amis*

# *Dédicaces*

On dédie ce travail :

A nos chers parents

Pour leur soutien, leur patience, et leur sacrifice, vous  
méritez tout éloge,

On espère être l'image que vous êtes fait de  
nous, que dieu vous garde et vous bénisse.

On dédie aussi ce travail à nos chers frères et  
sœurs, pour leur affection et leur encouragement qui  
ont toujours été pour nous des plus précieux.

Que ce travail soit pour vous le gage de notre profond  
amour.

A tout nos amis

A tous ceux qui nous ont aidés.

<b>INTRODUCTION GENERALE .....</b>		<b>1</b>
 <b>CHAPITRE I : MODELISATION ET ASSERVISSEMENT DE LA VITESSE D'UN MCC</b>		
I.1	Introduction.....	3
I.2	Définition.....	3
I.3	Description d'un moteur à courant continu .....	3
I.3.1	Les différents modes d'excitations .....	4
I.4	Modélisation du moteur à courant continu .....	6
I.5	Simulation en boucle ouverte du MCC .....	8
I.6	Asservissement du MCC avec un régulateur PI.....	9
I.7	Description des régulateurs PID.....	10
I.7.1	Action Proportionnelle .....	10
I.7.2	Action intégral .....	10
I.7.3	Action Dérivée .....	11
I.8	L'effet des paramètres d'un contrôleur PID sur la réponse .....	11
I.8.1	Proportionnelle.....	11
I.8.2	Intégral .....	11
I.8.3	Dérive.....	11
I.9	Synthèse, calcul et choix du correcteur.....	12
I.9.1	Etude de la boucle du courant .....	12
I.9.2	Etude de la boucle de la vitesse.....	14
I.9.3	Etude des deux boucles (courant et vitesse) .....	16
I.10	Conclusion .....	19
 <b>CHAPITRE II : COMMANDE DE LA MACHINE A COURANT CONTINUE PAR LES TECHNIQUE D'INTELLIGENCE ARTIFICIEL</b>		
II.1	Introduction.....	20
II.2	Intelligence artificielle .....	20
II.3	Techniques de Calcul intelligent .....	20
II.4	Algorithmes génétiques .....	21
II.4.1	Concepts et principe des algorithmes génétiques.....	21
II.4.1.1	La mise en œuvre d'un algorithme génétique.....	22
II.4.1.2	Vocabulaires des algorithmes génétiques .....	24

II.4.2	Le codage.....	24
II.4.3	La Population initiale.....	25
II.4.4	Fonction d'évaluation (fitness).....	25
II.4.5	Les opérateurs de base d'un AG.....	25
II.4.5.1	Le croisement.....	25
II.4.5.2	La mutation.....	26
II.4.6	La sélection.....	27
II.4.7	La convergence d'un AG.....	28
II.4.8	Développement d'un algorithme génétique pour le réglage PI.....	28
II.4.8.1	Ecrire la fonction objective pour un algorithme génétique..	28
II.4.8.2	Indices de performance .....	29
II.4.9	Application des AG à l'optimisation du régulateur PI pour la commande d'MCC.....	29
II.4.9.1	Boucle interne de régulation du courant .....	29
II.4.9.2	Boucle externe de régulation de vitesse.....	31
II.4.9.3	Application des paramètres du régulateur PI optimisés sur le modèle de la MCC.....	33
II.4.9.3	Conclusion .....	34
II.5	la logique floue.....	34
II.5.1.1	Historique de la logique floue .....	35
II.5.1.2	Premières applications .....	35
II.5.1.3	Développement .....	35
II.5.2	Avantages et inconvénients de la logique floue .....	35
II.5.3	Pourquoi utiliser la logique floue ?.....	36
II.5.4	Théorie de sous ensemble floue .....	36
II.5.4.1	Concept de sous-ensemble flou.....	36
II.5.4.2	Définition d'un sous-ensemble flou.....	37
II.5.4.3	Fonction d'appartenance.....	37
II.5.4.4	Opérations sur les sous ensembles flous .....	38
II.5.5	Commande par logique floue .....	40
II.5.5.1	Structure d'un régulateur flou.....	41
II.5.5.1.1	Base de connaissance .....	41
II.5.5.1.2	La fuzzification .....	42
II.5.5.1.3	Mécanisme d'inférence .....	42

II.5.6	Etapes de conception d'un contrôleur flou pour le réglage de la vitesse	46
II.5.6.1	Choix des variables d'entrées et de sorties.....	47
II.5.6.2	Normalisation des variables d'entrée et de sortie .....	47
II.5.6.3	Interface de fuzzification.....	47
II.5.6.4	Inférence.....	47
II.5.6.5	Traitement numérique de l'inférence.....	48
II.5.6.6	Défuzzification .....	48
II.5.7	Application de la commande floue sur le moteur à courant continu .....	48
II.5.7	Conception de contrôleur.....	49
II.5.7.2	Schéma bloc de simulation de la MCC avec le contrôleur floue	51
II.5.7.2	floue	52
II.5.8	Résultats de simulation et discussion.....	54
II.6	Conclusion.....	54
II.7	Comparaison entre les différentes méthodes de régulation	55
	Conclusion .....	

### **CHAPITRE III : OPTIMISATION PAR L'ALGORITHME PSO**

III.1	Introduction.....	56
III.2	Optimisation Par Essaims de Particule « PSO » .....	56
III.3	Notion de voisinage .....	56
III.4	L'algorithme PARTICLE SWARM OPTIMIZATION.....	57
III.4.1	Les variables et paramètres de l'algorithme .....	58
III.4.2	Initialisation de l'essaim .....	60
III.4.3	Confinement d'intervalle .....	60
III.4.4	Organigramme général d'un PSO.....	60
III.5	Avantages et inconvénients de PSO.....	61
III.6	Optimisation des Gains du PI par la Technique PSO.....	61
III.7	Résultats de simulation .....	63
III.8	Conclusion.....	64

**CONCLUSION GENERAL .....**

**Références bibliographiques .....**

## Liste des figures

<b>Figure I.1</b>	Fonctionnement de la machine a courant continu.....	<b>3</b>
<b>Figure I.2</b>	Schéma d'un moteur à courant continu.....	<b>4</b>
<b>Figure I.3</b>	Modélisation électrique d'un Moteur à excitation séparée.....	<b>4</b>
<b>Figure I.4</b>	Modélisation électrique d'un Moteur à excitation shunt.....	<b>5</b>
<b>Figure I.5</b>	Modélisation électrique d'un Moteur à excitation série.....	<b>5</b>
<b>Figure I.6</b>	Modélisation électrique d'un Moteur à excitation compound.....	<b>5</b>
<b>Figure I.7</b>	Schéma équivalent d'un MCC à excitation séparé .....	<b>6</b>
<b>Figure I.8</b>	Schéma bloc du Modèle du moteur électrique.....	<b>7</b>
<b>Figure I.9</b>	Réponse de la MCC en boucle ouverte .....	<b>8</b>
<b>Figure I.10</b>	Schéma bloc d'asservissement.....	<b>10</b>
<b>Figure I.11</b>	Schéma bloc de la boucle du courant.....	<b>12</b>
<b>Figure I.12</b>	Schéma bloc de la boucle du courant identifié .....	<b>12</b>
<b>Figure I.13</b>	Schéma bloc de la boucle de vitesse.....	<b>14</b>
<b>Figure I.14</b>	Schéma bloc de la boucle de la vitesse identifié .....	<b>14</b>
<b>Figure I.15</b>	Détermination du temps de réponse à 2%.....	<b>16</b>
<b>Figure I.16</b>	Schéma bloc de l'asservissement des deux boucles (vitesse et courant).....	<b>17</b>
<b>Figure I.17</b>	Régulation PI de la vitesse, de courant et couples .....	<b>18</b>
<b>Figure II.1</b>	Différentes méthodes de l'intelligence computationnelle .....	<b>21</b>
<b>Figure II.2</b>	Principe des algorithmes génétiques .....	<b>22</b>
<b>Figure II.3</b>	Organigramme de l'algorithme génétique.....	<b>23</b>
<b>Figure II.4</b>	Exemples d'opération de croisement.....	<b>26</b>
<b>Figure II.5</b>	Principe de la mutation binaire .....	<b>27</b>
<b>Figure II.6</b>	Schéma bloc de la boucle de courant.....	<b>30</b>
<b>Figure II.7</b>	Interface GATool pour l'asservissement du courant.....	<b>30</b>
<b>Figure II.8</b>	Réponse en courant à partir des paramètres du régulateur $PI_{AG}$ .....	<b>31</b>
<b>Figure II.9</b>	Schéma bloc de la boucle de vitesse.....	<b>31</b>
<b>Figure II.10</b>	Interface GATool pour l'asservissement de la vitesse.....	<b>32</b>
<b>Figure II.11</b>	Réponse en vitesse à partir des paramètres du régulateur $PI_{AG}$ .....	<b>33</b>
<b>Figure II.12</b>	Réponses en vitesse et en courant optimisées par l'AG.....	<b>33</b>
<b>Figure II.13</b>	Représentations des différents types.....	<b>38</b>
<b>Figure II.14</b>	Représentations de La hauteur, Le noyau, Le support.....	<b>38</b>
<b>Figure II.15</b>	Fonction d'appartenance A.....	<b>38</b>
<b>Figure II.16</b>	Fonction d'appartenance B.....	<b>38</b>

<b>Figure II.17</b>	Fonction d'appartenance de l'union.....	<b>39</b>
<b>Figure II.18</b>	Fonction d'appartenance d'intersection.....	<b>40</b>
<b>Figure II.19</b>	Opération de complémentation de la fonction d'appartenance A.....	<b>40</b>
<b>Figure II.20</b>	Structure de base d'un régulateur flou.....	<b>41</b>
<b>Figure II.21</b>	Définition de Alors dans la méthode d'inférence max-min.....	<b>43</b>
<b>Figure II.22</b>	Définition de OU dans la méthode d'inférence max-min.....	<b>43</b>
<b>Figure II.23</b>	Fonction d'appartenance résultante.....	<b>44</b>
<b>Figure II.24</b>	Définition de ALORS dans la méthode d'inférence max-prod.....	<b>44</b>
<b>Figure II.25</b>	Définition de OU dans la méthode d'inférence.....	<b>45</b>
<b>Figure II.26</b>	Méthode par centre de gravité.....	<b>47</b>
<b>Figure II.27</b>	Méthode du maximum.....	<b>47</b>
<b>Figure II.28</b>	Définition de la structure globale.....	<b>49</b>
<b>Figure II.29</b>	Définition des fonctions d'appartenances pour 'e', 'de/dt' et 'u'.....	<b>50</b>
<b>Figure II.30</b>	Écran de saisie des règles .....	<b>51</b>
<b>Figure II.31</b>	Schéma bloc d'asservissement de la vitesse par RLF.....	<b>52</b>
<b>Figure II.32</b>	Réponses en vitesse et en courant régulées par logique floue .....	<b>53</b>
<b>Figure II.33</b>	Allure finale de la vitesse et du courant (Comparaison entre les différentes méthodes PI, BLF, AG).....	<b>55</b>
<b>Figure III.1</b>	Schéma de principe du déplacement d'une particule.....	<b>58</b>
<b>Figure III. 2</b>	Organigramme général de l'PSO.....	<b>60</b>
<b>Figure III.3</b>	Schéma bloc de simulation de la MCC avec le PSO.....	<b>61</b>



## Liste des Tableaux

Tableaux I.1	Caractéristiques de la MCC.....	8
Tableaux I.2	Influence des paramètres $K_p$ , $K_i$ et $K_d$ sur la réponse du système....	12
Tableaux II.3	Règles à cinq fonctions d'appartenance.....	48

## **Liste des abréviations**

**MCC** : Machine à Courant Continu

**PID** : Proportionnel intégral Dérive

**PI** : Proportionnel intégral

**P** : Proportionnel

**I** : intégral

**D** : Dérive

**K<sub>p</sub>** : Gain proportionnel

**K<sub>i</sub>** : Gain d'intégration

**IA** : intelligence artificielle

**AG** : Algorithmes génétiques

**ITAE** : Intégrale du temps multipliée par l'erreur absolue

**IAE** : Intégrale de la grandeur absolue de l'erreur

**ISE** : Intégrale du carré de l'erreur

**MSE** : Moyenne du carré de l'erreur

**RLF** : Régulateur logique floue

**NG** : Négatif Grand

**N**: Négatif

**Z** : Zéro

**P**: Positif

**PG** : Positif Grand

**FIS** : Fuzzy inference system

**PSO** : L'optimisation par essaim particulaire



# *Introduction générale*

### Introduction générale :

La régulation est l'un des volets de l'automatique qui connaît un essor considérable vu son importance dans l'industrie.

Au tout début, on faisait appel à la régulation classique à cause de ses avantages comme sa simplicité à utiliser, sa fiabilité et son efficacité, les méthodes classiques de régulation basées sur le dimensionnement du régulateur en utilisant le modèle mathématique du système, mais souvent les qualités d'un régulateur classique restent limitées à cause de la complexité réelle des systèmes à commander (non linéarité) et plus d'inconvénients dans le milieu industriel.

Plusieurs applications industrielles nécessitent l'utilisation des méthodes modernes de contrôle, permettant d'avoir une réponse rapide et des performances dynamiques élevées. Parmi ces méthodes, on cite le réglage par la logique floue (RLF), les algorithmes génétiques (AG) et aussi l'optimisation par essais particuliers (PSO).

Les concepts de ces deux approches (RLF, AG) sont basés sur la perception globale qu'un être humain a de son environnement qui l'avoisine pour la logique floue, et pour les algorithmes génétiques, sur une tentative de modéliser le comportement 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), et l'objectif de PSO est trouver l'optimum global de quelque multidimensionnel (habituellement non linéaire) fonction. L'algorithme a prouvé efficace dans résoudre beaucoup de problèmes.

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 méthodes et à les mettre en œuvre pour la commande de la vitesse d'une machine à courant continu, et ce dans le but de comparer leurs performances avec celles utilisant des techniques conventionnelles.

Dans le cadre de notre travail, nous allons étudier un moteur à courant continu commandé par l'intelligence artificielle et réaliser pratiquement un asservissement de vitesse de MCC. Cette mémoire est donc constituée d'une introduction générale, de trois chapitres et d'une conclusion générale.

## Introduction générale

---

Cette mémoire est présentée en trois chapitres

Le premier chapitre est consacré pour présenter la modélisation de la machine et puis déterminer la commande PI.

Dans le deuxième chapitre, nous détaillerons les formalismes de base de la logique floue et des algorithmes génétiques et la méthodologie adoptée pour la conception du système de commande basé sur ces deux approches. L'application de ces commandes et l'étude comparative des performances des systèmes étudiés seront présentées.

Le troisième chapitre, on a étudiés la commande de vitesse d'un moteur à courant continu par l'optimisation par Essaim de particule pour l'objectif d'avoir des résultats performances

En fin ce travail sera clôturé par une conclusion générale résumant les différents résultats obtenus.

# Chapitre I

**Modélisation et Asservissement de la vitesse  
d'une MCC**

## I.1/ Introduction

Les moteurs à courant continu sont très utilisés dans les systèmes automatiques qui nécessitent une variation précise de la vitesse de rotation.

Dans ce chapitre, on présente la constitution de la machine à courant continu et leur principe de fonctionnement. Il s'agit d'établir les différents types des moteurs à courant continu. Ensuite nous donnerons les avantages et les inconvénients Enfin le domaine d'utilisation.

## I.2/ Définition

Les machines à courant continu sont des convertisseurs électromécaniques d'énergie : Soit ils convertissent l'énergie électrique absorbée en énergie mécanique lorsqu'ils sont capables de fournir une puissance mécanique suffisante pour démarrer puis entraîner une charge en mouvement. On dit alors qu'ils ont un fonctionnement en moteur. Soit ils convertissent l'énergie mécanique reçue en énergie électrique lorsqu'ils subissent l'action d'une charge entraînée. On dit alors qu'ils ont un fonctionnement en générateur.

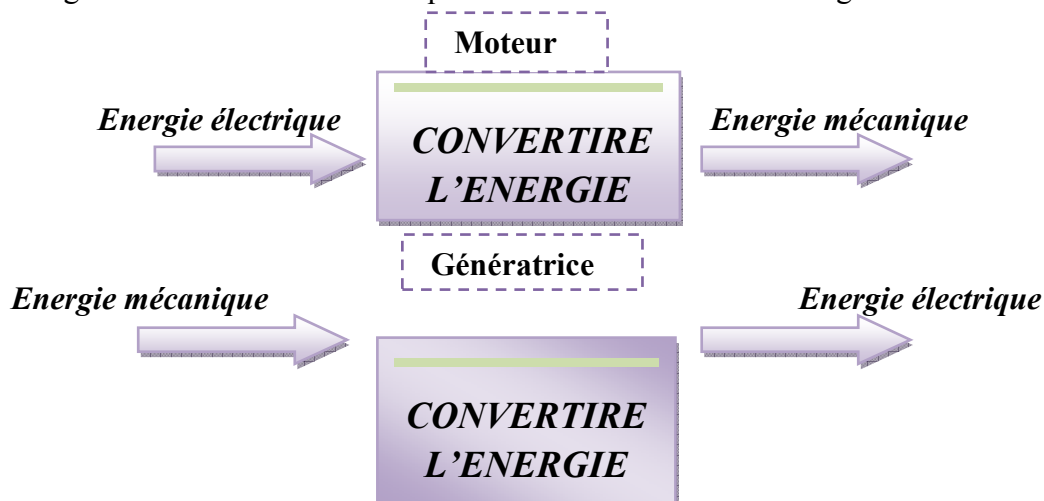


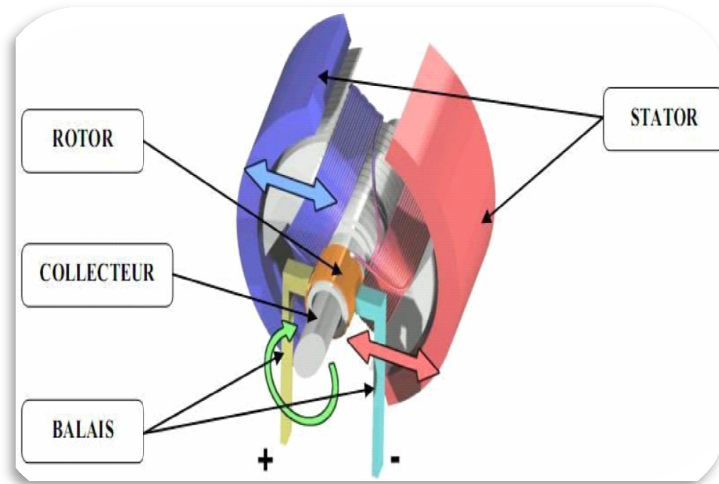
Figure. I.1 : Fonctionnement de la machine à courant continu

## I.3/Description d'un moteur à courant continu

Un moteur électrique à courant continu est constitué :

- D'un stator qui est à l'origine de la circulation d'un flux magnétique longitudinal fixe créé soit par des enroulements saturés (bobinage) soit par des aimants permanents à l'arrière du stator, se trouve la partie porte balais et les balais assurant les contacts électriques avec le rotor. Il est aussi appelé inducteur [1].
- D'un rotor bobiné relié à un collecteur rotatif inversant la polarité dans chaque enroulement rotorique au moins une fois par tour de façon à faire circuler un flux

magnétique transversal en quadrature avec le flux statorique. Les enroulements rotoriques sont aussi appelés enroulements d'induits, ou communément induit.



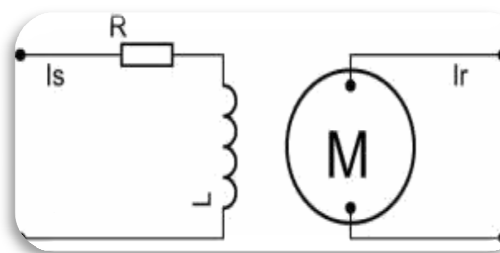
**Figure. I.2 :** Schéma d'un moteur à courant continu

### I.3.1/ Les différents modes d'excitations

Les moteurs à courant continu se différencient par la manière dont on fournit le courant d'excitation. Les différents cas possibles sont :

#### a) Moteur à excitation séparée

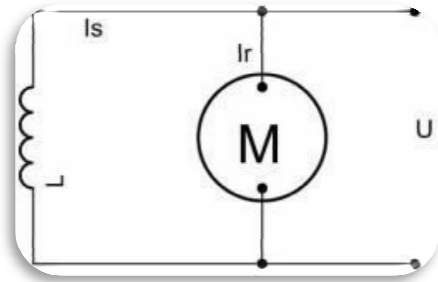
Ce mode d'excitation nécessite deux sources d'alimentations distinctes. L'alimentation de l'enroulement inducteur est prise sur une source indépendante de la source principale. On change le sens de rotation en permutant les bornes de l'induit ou de l'inducteur. Le circuit électrique est représenté par la suivante :



**Figure. I.3 :** Modélisation électrique d'un Moteur à excitation séparée

#### b) Moteur à excitation shunt (Dérivée)

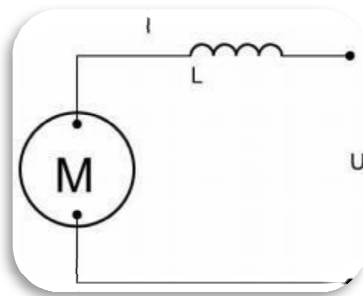
L'enroulement d'excitation est connecté en parallèle sur l'alimentation du moteur, il possède les mêmes propriétés que le moteur à excitation séparée du fait que, dans les deux cas, l'inducteur constitue un circuit extérieur à celui de l'induit. Le circuit électrique est représenté par la suivante :



**Figure. I.4 :** Modélisation électrique d'un Moteur à excitation shunt

**c) Moteur à excitation série**

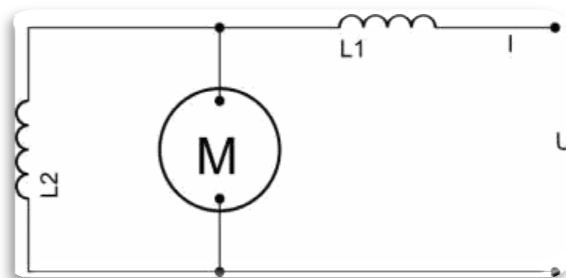
Le circuit d'excitation est placé avec l'induit du moteur. Sa particularité est d'avoir un inducteur qui est traversé par le même courant, l'inducteur possède donc une résistance plus faible que celle des autres types de machines. L'inducteur est en série avec l'induit : une seule source d'alimentation suffit. On change le sens de rotation en permutant les connexions de l'induit et de l'inducteur. Le circuit électrique est représenté par la suivante :



**Figure. I.5 :** Modélisation électrique d'un Moteur à excitation série

**d) Moteur à excitation compound (Composée) :**

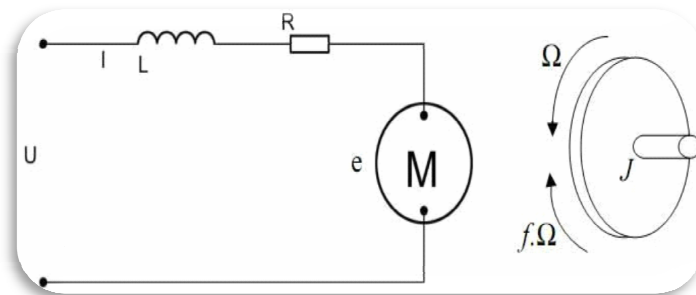
C'est une combinaison des excitations Shunt et Série, on peut réaliser un compound-âge additif (Si les flux des deux enroulements s'additionnent) ou soustractif (dans le cas contraire).



**Figure. I.6:** Modélisation électrique d'un Moteur à excitation compound

## I.4/ Modélisation du moteur à courant continu

Le moteur à courant continu peut être modélisé par le biais d'équations électriques, électromécaniques et mécaniques. Ces trois groupes d'équations nous permettront de mieux appréhender le moteur à courant continu dans son fonctionnement réel. Du côté électrique, nous pouvons dire que le moteur à courant continu se définit par un circuit induit et un circuit inducteur; l'induit de la MCC peut être vu comme une résistance et une inductance en série avec une source de tension commandée et proportionnelle à la vitesse [2]. Du côté mécanique, nous représentons la machine à courant continu par l'inertie de l'induit augmentée de celui de la charge entraînée. La Figure I.7 montre le schéma équivalent du moteur à courant continu.



**Figure I.7 :** Schéma équivalent d'un MCC à excitation séparé

L'équation électrique, liant la tension  $V(t)$  aux bornes de l'induit (rotor), le courant d'induit  $i(t)$  et l'équation (I.1)

a force électromotrice ( $e(t)$ ) est donnée par :

$$Ri(t) + L \frac{di(t)}{dt} + e(t) = V(t) \quad (\text{I.1})$$

Où :

$R$  : la résistance de l'induit du MCC

$L$  : inductance

$e(t)$  : la force électromotrice qui est proportionnelle à la vitesse de rotation du rotor :

$$e(t) = K_e \Omega(t) \quad (\text{I.2})$$

Où :  $K_e$  : est la constante électrique du moteur (constante de vitesse)

$\Omega(t)$  : la vitesse de rotation.

L'équation mécanique rendant compte des couples agissant sur le rotor s'écrit :

$$J \frac{d\Omega(t)}{dt} = C(t) - C_r(t) - f\Omega(t) \quad (\text{I.3})$$

Où :  $C(t)$  : est le couple moteur

$C_r(t)$  : le couple résistant (charge et perturbations)

$f$  : le coefficient de frottement visqueux

$J$  : le moment d'inertie du rotor.

Par construction, le couple  $c(t)$  est proportionnel au courant d'induit  $i(t)$  :

$$C(t) = k_c (t) \quad (\text{I.4})$$

Où :  $k_c$  est la constante du couple moteur

En général, les coefficients  $K_e$  et  $K_c$  sont si proches qu'il est raisonnable de les considérer égaux, négligeant alors les pertes durant la conversion électromécanique de puissance.

On pose :  $K = K_e = K_c$

Les transformées de Laplace des équations (I.1), (I.2), (I.3) et (I.4) sont données par :

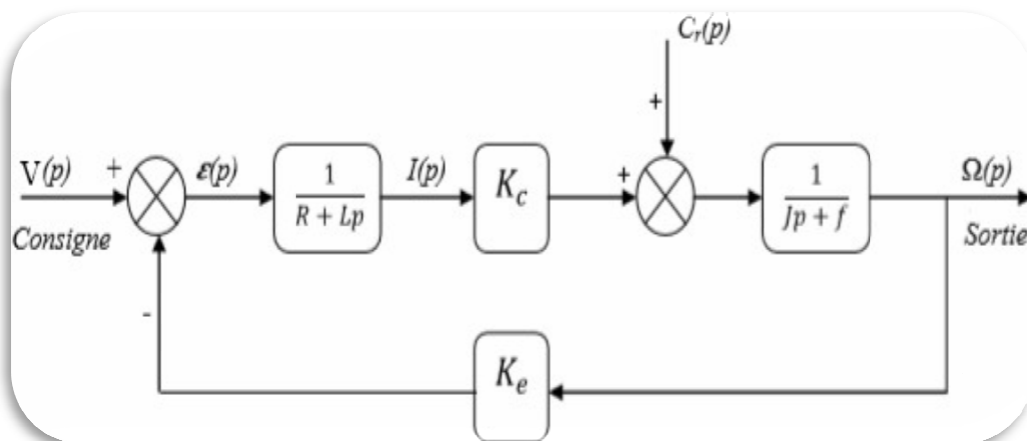
$$(R + Lp)(P) = V(P) - E(P) \quad (\text{I.5})$$

$$E(P) = k \Omega(P) \quad (\text{I.6})$$

$$C(P) - C_r(P) = (f + Jp)\Omega(P) \quad (\text{I.7})$$

$$C(P) = kI(P) \quad (\text{I.8})$$

Les transformées de Laplace obtenues nous permettent de modéliser le moteur sous forme de schéma bloc ou schéma fonctionnel. Ces schémas présentent les fonctions de transfert suivant les paramètres d'entrée et de sortie considérés. Si on considère comme grandeur d'entrée la tension appliquée aux bornes de l'induit et comme grandeur de sortie la vitesse du moteur, nous obtenons le schéma bloc tension- vitesse. Ce dernier nous permet de voir l'évolution de la vitesse avec la tension, qui est donné par la Figure I.8



**Figure. I.8** : Schéma bloc du Modèle du moteur électrique

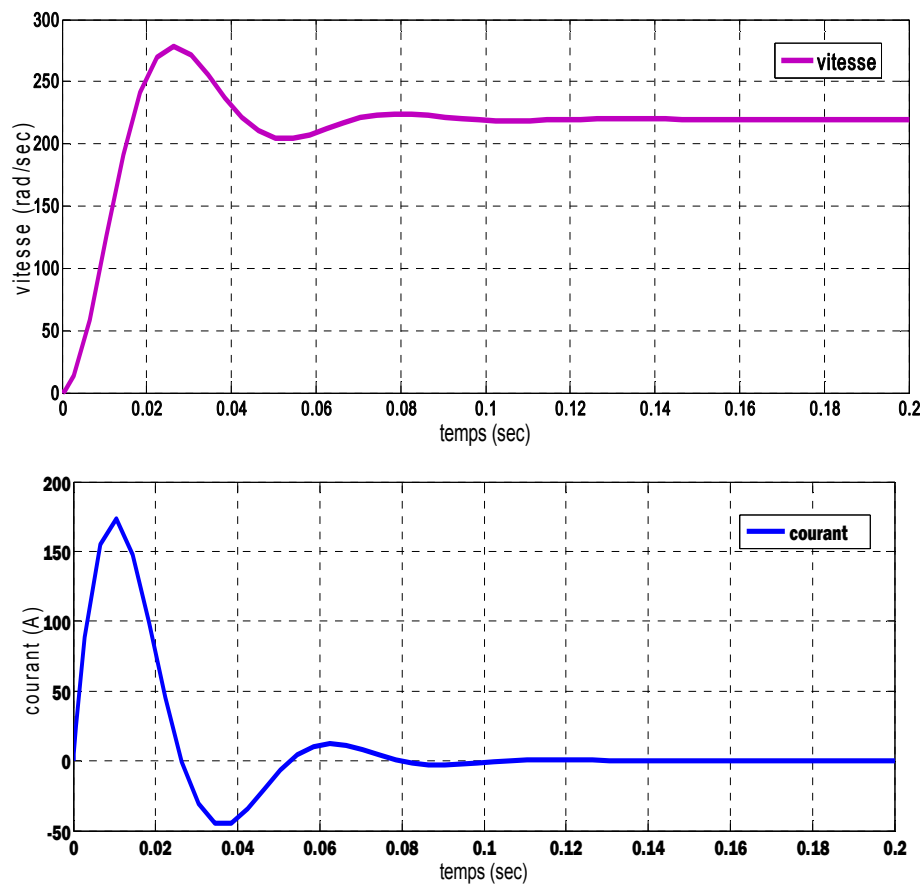
## I.5 / Simulation en boucle ouverte du MCC

Les caractéristiques du Moteur à courant continu sont données par le tableau I.1

Tension nominale	$V=220\text{ V}$
Vitesse à la tension nominale	$N=2100\text{ tr/mn}$
Résistance de l'induit	$R=0.6\ \Omega$
Inductance de l'induit	$L=6\text{ mH}$
Constante de vitesse et de couple	$K=1\text{ V/rad/s}$
Coefficient de frottements visqueux	$f=0.001\text{ N.m/rd. s}^{-1}$
Inertie du moteur	$J=0.01\text{ Kg.m}^2$

**Tableaux I.1** : Caractéristiques de la MCC.

Les résultats obtenus sont illustrés par la Figure I.9



**Figure I.9** : Réponse de la MCC en boucle ouverte.

### ✚ Interprétation

D'après la figure précédente on constate que : La réponse de vitesse oscille avec un dépassement=13.5%. Et un temps de réponse  $\tau_{\text{réponse}} = 0.065 \text{ sec}$ , et un temps de montée  $r_{\text{éponse}} = 0.015 \text{ sec}$ . Le courant subit au moment du démarrage un pic, puis atteint la valeur nulle.

Pour un moteur électrique à courant continu, on a constaté que la réponse en vitesse est instable, d'où la nécessité d'une commande pour stabiliser le système et améliorer ses performances.

## I.6/ Asservissement du MCC avec un régulateur PI

L'asservissement est réalisé à l'aide d'un régulateur Proportionnel-Intégral-Dérivée PID qui se compose de trois termes P, I et D, d'où le P correspond au terme proportionnel, I pour le terme intégral et D pour le terme dérivé. Le régulateur PID est très utilisé pour l'asservissement des moteurs à courant continu soit en position ou en vitesse. On s'intéresse dans ce travail par l'asservissement en vitesse du MCC. La loi de commande est donnée par l'équation suivante :

$$(t) = kp(t) + ki \int_0^t \varepsilon(t)dt + kd d\varepsilon dt (t) \quad (I.9)$$

avec

$u$  : désigne la commande.

$\varepsilon$  : désigne l'erreur entre la consigne Entrée E et la sortie réelle S.

$kp, ki$  et  $ka$  sont les paramètres du contrôleur PID .

Le régulateur PID est inséré dans la chaîne directe de l'asservissement, en série avec le processus, comme est indiqué dans la Figure I.10. Ce régulateur élabore, à partir du signal d'erreur  $\varepsilon (t)$ , une commande  $u(t)$  en fonction de trois actions. Il a la capacité d'éliminer la compensation de l'état d'équilibre grâce à l'action intégrale, et il peut anticiper le futur grâce à une action dérivée [4]. Ce travail a pour but, d'implémenter la commande PID classique pour un MCC. L'objectif est d'annuler l'erreur statique (Précision), de diminuer le dépassement (Stabilité), et de diminuer le temps de réponse et le temps de montée (Rapidité). L'ajustement des paramètres  $kp, ki$  et  $kid$  de la commande PID est basé sur la méthode classique.

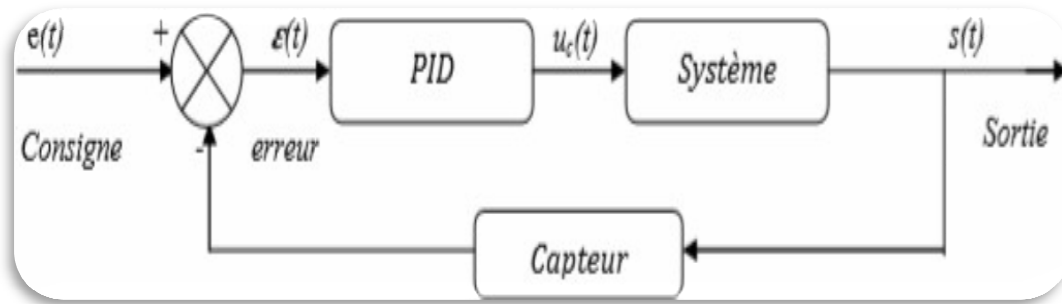


Figure I.10 : Schéma bloc d'asservissement.

## I.7/ Description des régulateurs PID

Un régulateur PID remplit essentiellement trois fonctions :

- Il fournit un signal de commande  $u(t)$  en tenant compte de l'évolution du signal de sortie  $s(t)$  par rapport à la consigne  $e(t)$
- Il élimine l'erreur statique grâce au terme intégrateur.
- Il anticipe les variations de la sortie grâce au terme dérivateur.

Le PID représente les abréviations des trois actions qu'il utilise pour effectuer ses corrections, ce sont des ajouts d'un signal à un autre. Tous agissent sur la quantité régulée.

Les actions aboutissent finalement à des soustractions de l'erreur de mesure.

### I.7.1/ Action Proportionnelle

Pour cette action, l'erreur est multipliée par la constante P (pour Proportionnel) qui est négative, puis ajoutée (soustraction de l'erreur de mesure) à la quantité régulée. P est valide uniquement sur la bande dans laquelle le signal de sortie du régulateur est proportionnel à l'erreur du système. Noter que si l'erreur de mesure est égale à zéro, la partie proportionnelle de la sortie du régulateur est également à zéro [6].

### I.7.2/ Action Intégrale

L'action intégrale fait intervenir la notion de temps. Elle tire profit du signal d'erreur passé qui est intégré (additionné) pendant un laps de temps, puis multiplié par la constante I (négative) ce qui en fait une moyenne, elle est enfin additionnée (soustraction de l'erreur de mesure) à la quantité régulée. La moyenne de l'erreur de mesure permet de trouver l'erreur moyenne entre la sortie du régulateur et la valeur de la consigne. Un système seulement proportionnel oscille en plus et en moins autour de la consigne du fait qu'en arrivant vers la consigne, l'erreur est à zéro, il n'enlève alors plus rien est dépassé la consigne, ou oscille et/ou se stabilise à une valeur trop basse ou trop élevée. L'addition sur l'entrée d'une proportion négative (soustraction) de l'erreur de mesure moyennée, permet toujours de

réduire l'écart moyen entre la mesure en sortie et la consigne. Donc finalement, une boucle PI bien réglée verra sa sortie redescendre lentement à la valeur de la consigne [6].

### I.7.3/ Action Dérivée

L'action dérivée utilise aussi la notion de temps. Elle cherche à anticiper l'erreur future. La dérivée première (la pente de l'erreur) est calculée pour un laps de temps et multipliée par la constante (négative) D, puis elle est additionnée (soustraction de l'erreur de mesure) à la quantité régulée. L'action dérivée de la régulation fournit une réponse aux perturbations agissant sur le système. Plus important est le terme dérivé, plus rapide sera la réponse en sortie à une perturbation sur l'entrée.

## I.8/ L'effet des paramètres d'un contrôleur PID sur la réponse

### I.8.1/ Proportionnel

L'erreur est multipliée par une constante  $K_p$  :

$$(t) = k\varepsilon(t) \quad (\text{I.10})$$

Plus  $K_p$  est grand, plus que la réponse est rapide et plus que l'erreur statique est corrigé.

### I.8.2/ Intégral

L'erreur est intégrée sur un intervalle de temps, puis multipliée par une constante  $K_i$  :

$$(t) = k_i \int_0^t \varepsilon(t) dt \quad (\text{I.11})$$

Plus  $K_i$  est élevé, plus que l'erreur statique est corrigée.

### I.8.3/ Dérive

L'erreur est dérivée par rapport au temps, puis multipliée par une constante  $K_d$  :

$$(t) = k_d \frac{d\varepsilon(t)}{dt} \quad (\text{I.12})$$

Réduit le dépassement et le temps de stabilisation [7] lorsque :

- $K_p$  augmente → montée plus rapide mais plus de dépassement.
- $K_i$  augmente → montée plus rapide mais régime stationnaire plus long, erreur statique plus faible.
- $K_d$  augmente → diminue le dépassement et le temps d'établissement du régime stationnaire, mais augmente la sensibilité au bruit

Le tableau suivant résume l'effet de chaque paramètre de contrôleur PID sur la réponse

Coefficient	Temps de Monté	Temps de stabilisation	Dépassement	Erreur statique
$K_P$	Diminue	Augmente	Augmente	Diminue
$K_I$	Diminue	Augmente	Augmente	Annule
$K_D$	-	Diminue	Diminue	-

**Tableau I.2:** Influence des paramètres  $K_p$ ,  $K_i$  et  $K_d$  sur la réponse du système

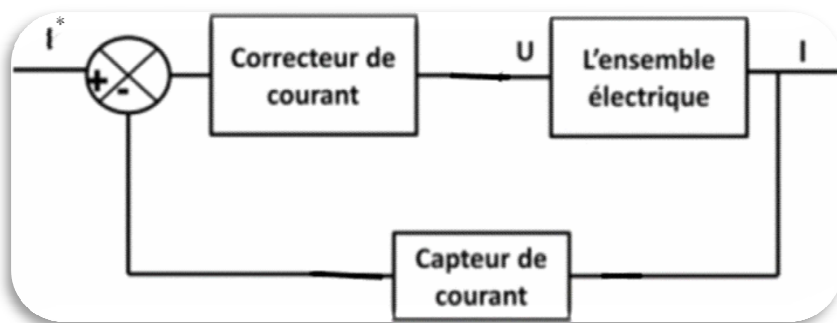
## I.9/ Synthèse, calcul et choix du correcteur

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 supposée constante pendant que les courants sont régulés.

Régler une boucle de régulation consiste à agir sur les paramètres des différentes actions (gain du proportionnel, gain de l'intégral, gain de la dérivée) sur des valeurs optimales pour obtenir la réponse désirée sur la sortie du procédé [6].

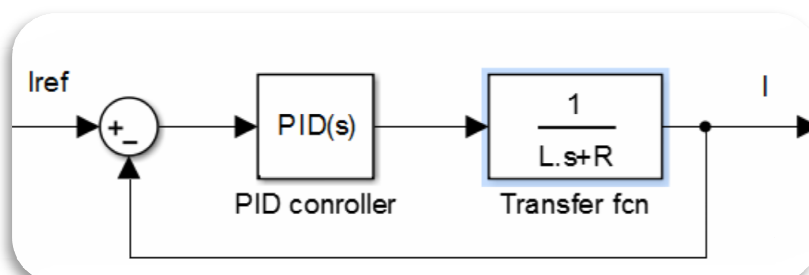
### I.9.1/ Etude de la boucle du courant

La boucle de régulation du courant peut se représenter par le schéma bloc suivant:



**Figure I.11 :** Schéma bloc de la boucle du courant

#### a) Identification des blocs



**Figure I.12:** Schéma bloc de la boucle du courant identifié

Avec : La fonction de transfert du courant est :

$$FT_{\text{électrique}} = \frac{1}{L.p + R} = \frac{1/R}{\frac{L}{R}p + 1} = \frac{1/R}{\tau p + 1} \quad (\text{I.13})$$

La fonction de transfert du correcteur PI est :

$$PI = K_P + \frac{K_I}{p} = \frac{K_P p + K_I}{p} = \frac{K_I(1 + \frac{K_P}{K_I}.P)}{p} = \frac{K_I(1 + \tau_{reg}.P)}{p} \quad (\text{I.14})$$

Et un retour unitaire égal 1

Notons :  $\tau_{reg} = \frac{K_P}{K_I}$  et  $\tau_{elec} = \frac{L}{R}$ .

### b) Asservissement du courant

#### • Boucle ouverte :

La fonction de transfert en boucle ouverte est la suivante:

$$FTBO(p) = PI \times FT_{\text{électrique}} = \frac{K_I(1 + \tau_{reg}.P)}{p} \times \frac{1/R}{\tau_{elec}p + 1} \quad (\text{I.15})$$

On a choisir

$\tau_{elec} = \tau_{reg}$  pour éliminer le dénominateur de la fonction de transfert de l'ensemble électrique.

Donc :

$$FTBO(p) = \frac{K_I/R}{p} \quad (\text{I.16})$$

Notons :  $\frac{K_I}{R} = K_I$

#### • Boucle fermée:

La fonction de transfert en boucle fermée est la suivante:

$$FTBF(P) = \frac{FTBO}{1 + FTBO} = \frac{\frac{K_1}{P}}{1 + \frac{K_1}{P}} = \frac{K_1}{K_1(\frac{1}{K_1}P + 1)} = \frac{1}{1 + \tau_{BF}P} \quad (\text{I.17})$$

Notons :

$$\tau_{BF} = \frac{1}{K_1}$$

Selon le cahier de charge, on veut que le temps de réponse en boucle fermée corrigée soit rapide 20 fois par rapport à la boucle ouverte non corrigée.

On pose :

$$\tau_{BF} = \frac{\tau_{Bo}}{20} = \frac{1}{K_1} = \frac{1}{K_I/R} = \frac{R}{K_I} = \frac{\tau_{Bo}}{20} \quad (\text{I.18})$$

$$K_I = \frac{20R}{\tau_{Bo}}$$

$$\tau_{reg} = \frac{K_P}{K_I} \Rightarrow K_P = \tau_{reg} \times K_I = \tau_{elec} \times K_I = K_I \times \frac{L}{R}$$

$$K_P = K_I \times \frac{L}{R} \tag{I.19}$$

$$\tau_{rBF} = 3\tau_{BF} = \frac{3\tau_{sys}}{20} = \frac{R}{K_I} = \frac{3L}{20R} = \frac{R}{K_I}$$

Application numérique :

$$K_I = \frac{20R^2}{3L} = \frac{20 \times (0.6)^2}{3(0.006)} = 400 \tag{I.20}$$

$$K_P = K_I \times \frac{L}{R} = 400 \times \frac{0.006}{0.6} = 4 \tag{I.21}$$

Les résultats obtenus de régulateur PI du courant est :

$$PI = K_P + \frac{K_I}{P} = 4 + \frac{400}{P} \tag{I.22}$$

### I.9.2/ Etude de la boucle de la vitesse

Il s'agit d'étudier la boucle de vitesse sans implémenter la boucle de courant. Cet asservissement a pour schéma bloc:

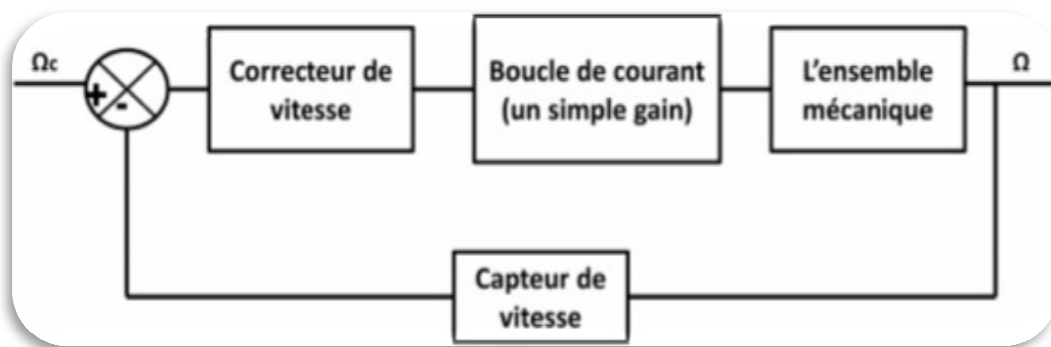


Figure I.13 : Schéma bloc de la boucle de vitesse

a) Identification des blocs:

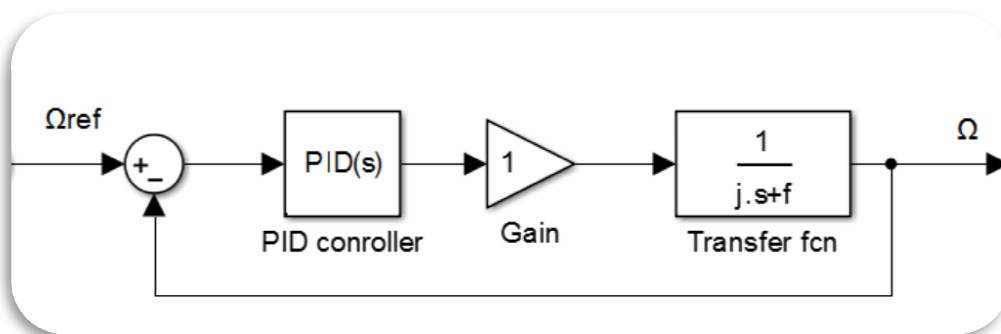


Figure I.14 : Schéma bloc de la boucle de la vitesse identifié

Avec : La fonction de transfert de la vitesse est :

$$FT_{mec} = \frac{1}{J.p + f} \tag{I.23}$$

La fonction de transfert du correcteur PI est :

$$PI = K_P + \frac{K_I}{P} = \frac{K_P P + K_I}{P} = \frac{K_I(1 + \frac{K_P}{K_I} K_I)}{P} \quad (I.24)$$

Et un retour unitaire égal 1

Notons :

$$K_2 = \frac{K_P}{K_I} \quad ; \quad K_V = K_I$$

### b) Asservissement de la vitesse

#### • Boucle ouverte :

La fonction de transfert en boucle ouverte est la suivante :

$$FT_{BO}(p) = PI \times FT_{mec} = \frac{K_V(1 + K_2 P)}{P} \times \frac{1}{jp + f} \quad (I.25)$$

$$FT_{BO} = \frac{K_V(1 + K_2 P)}{P(JP + f)} \quad (I.26)$$

#### • Boucle fermée:

La fonction de transfert en boucle fermée est la suivante:

$$FT_{BF} = \frac{FT_{BO}}{1 + FT_{BO}} = \frac{\frac{K_V(1 + K_2 P)}{P(JP + f)}}{1 + \frac{K_V(1 + K_2 P)}{P(JP + f)}} = \frac{K_V(1 + K_2 P)}{P(Jp + f) + K_V(1 + K_2 P)} = \frac{K_V(1 + K_2 P)}{JP^2 + (f + K_V K_2)P + K_V} \quad (I.27)$$

En imposant toujours des pôles correspondants au critère utilisé

$$\frac{K_V(1 + K_2 P)}{JP^2 + (f + K_V K_2)P + K_V} = \frac{\omega_n^2}{P^2 + 2\xi\omega_n P + \omega_n^2} \quad (I.28)$$

Cette fonction peut s'écrire sous d'autres formes :

$$\frac{K_V(1 + K_2 P)}{JP^2 + \frac{(f + K_V K_2)P}{J} + \frac{K_V}{J}} = \frac{\omega_n^2}{P^2 + 2\xi\omega_n P + \omega_n^2} \quad (I.29)$$

L'équation terme à terme de l'équation (I.27) entraîne que :

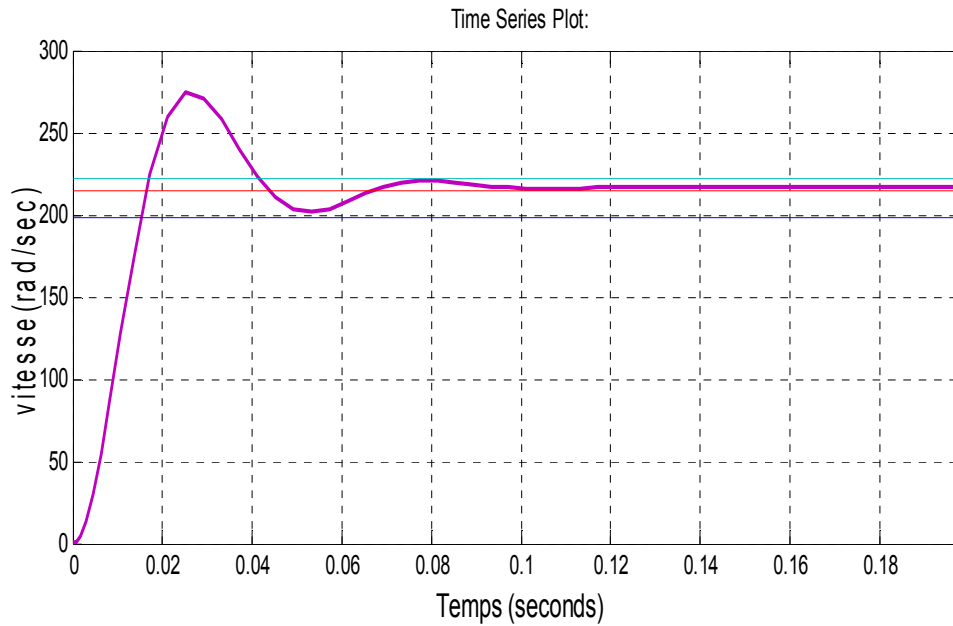
$$\left\{ \begin{array}{l} 2\xi\omega_n = \frac{f + K_V K_2}{J} \\ \omega_n^2 = \frac{K_V}{J} \end{array} \right. \Rightarrow \left\{ \begin{array}{l} K_V = \omega_n^2 \times J \\ K_2 = \frac{J2\xi\omega_n - f}{K_V} \end{array} \right.$$

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

- Un amortissement  $\xi=1$
- Une pulsation  $\omega_n$  a temps de réponse a 2%

#### • Détermination du temps de réponse à 2% d'un système 2ème ordre

La valeur de temps de réponse a 2% de la valeur final



**Figure I.15** : Détermination du temps de réponse à 2%

$$\tau_{reponse} = \frac{4}{\xi \omega_n} \Rightarrow \xi = 1 \text{ donc } \omega_n = \frac{4}{\tau_{reponse}} \quad (\text{I.31})$$

A partir de simulation :  $\tau_{reponse} = 0.0653 \text{ sec}$

$$\text{donc : } \omega_n = \frac{4}{\tau_{reponse}} = \frac{4}{0.0653} = 61.25 \quad (\text{I.32})$$

Application numérique :

$$k_v = k_I = \omega_n^2 \times J = 61.25 \times 0.01 = 37.51$$

$$k_2 = \frac{J2\xi\omega_n - f}{k_v} = \frac{0.01 \times 1 \times 61.25 - 0.001}{37.51} = 0.0331 \quad (\text{I.33})$$

$$k_2 = \frac{k_p}{k_I} \Rightarrow k_p = k_2 \times k_I = 0.0331 \times 37.51 \quad (\text{I.34})$$

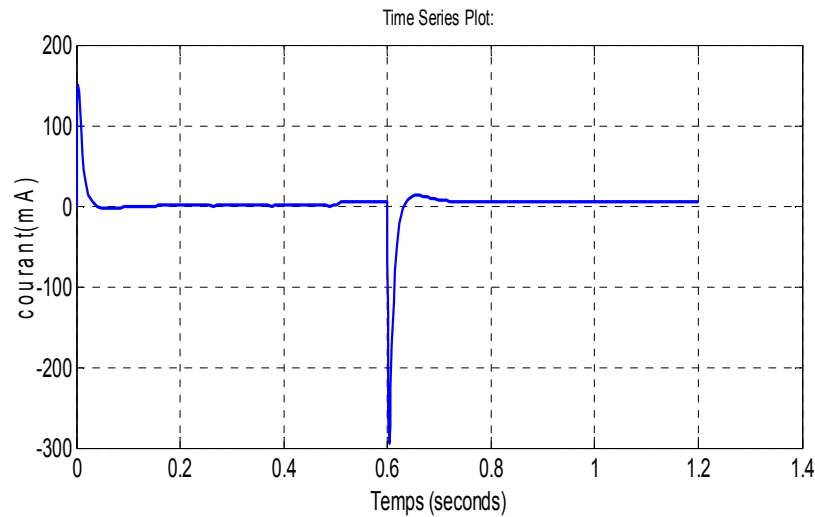
Les résultats obtenus de régulateur PI de la vitesse est :

$$PI = k_p + \frac{k_p}{p} = 1.244 + \frac{37.51}{p} \quad (\text{I.35})$$

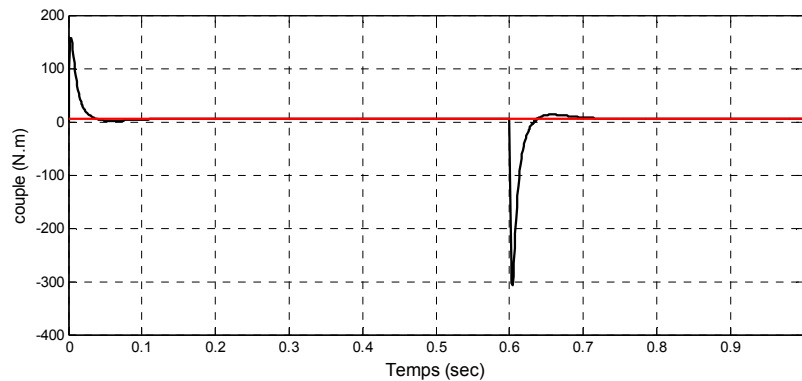
### I.9.3/ Etude des deux boucles (courant et vitesse)

De ce qui précède, on déduit le schéma complet de contrôle d'un moteur à CC, à l'aide d'une régulation cascade incluant deux fonctions de transfert. Comme la constante de temps électrique est plus faible que la constante de temps mécanique, la boucle interne (qui





c) Allure du courant



d) Allure des couples

Figure I.17 : Régulation PI de la vitesse, de courant et couples

- **Interprétation :**

D'après la figure précédente on constate que : La réponse de vitesse atteint la consigne (position désirée) en un temps de réponse  $\tau_{\text{réponse}} = 0.015 \text{ sec}$ , et un temps de montée  $\tau_{\text{réponse}} = 0.013 \text{ sec}$ , avec un dépassement=13.5%. On applique une charge après 0.3sec la vitesse chute de 3 rad/sec et retourne à sa valeur initiale avec un temps de rejet égal 0.08 sec. Le courant subit au moment du démarrage un pic, puis atteint rapidement la valeur nulle. On remarque qu'au moment d'application de la charge, le couple a augmenté pour équilibrer la charge (couple résistant)

**I.10/ Conclusion**

Dans ce chapitre, on a présenté une généralité sur la MCC et ses différents types d'excitation. Ensuite on a abordé sa modélisation, on peut conclure que la commande PI appliquée aux moteurs électriques à courant continu était un régulateur qui pouvait offrir de bonnes performances performance, que ce soit au niveau de la précision, du temps de réponse, et de la stabilité.

# Chapitre II

Commande d'un MCC par l'intelligence  
artificielle

## II.1/ Introduction

L'intelligence artificielle est « l'ensemble des théories et des techniques mises en œuvre en vue de réaliser des machines capables de simuler l'intelligence »

Ce chapitre, consiste à étudier les deux approches principales de l'intelligence artificielle (logique floue et algorithme génétique) et à les mettre en œuvre pour la commande de la vitesse d'une machine à courant continu.

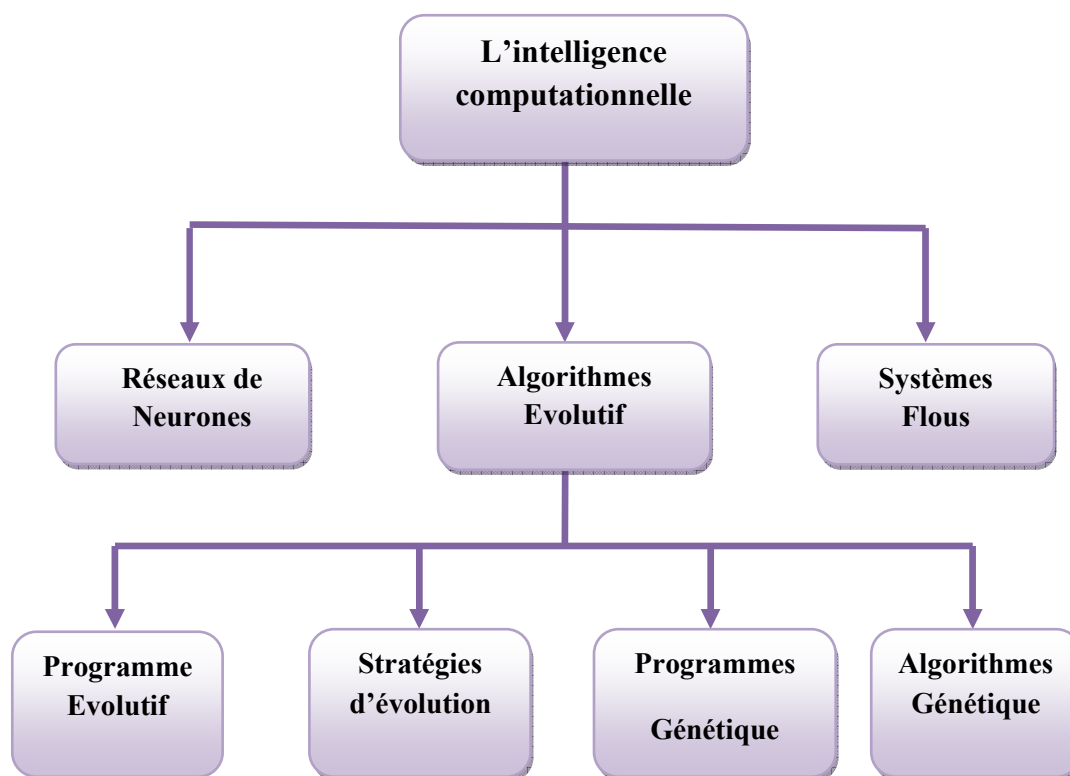
## II.2/ L'Intelligence Artificielle

L'Intelligence Artificielle (IA) appartient au domaine de l'informatique et concerne les approches informatiques ayant pour but de faire apparaître un comportement intelligent. C'est une science qui s'intéresse à la conception de machines pouvant simuler la cognition humaine. En effet, elle tend à donner à la machine un comportement "humain", capable notamment de s'adapter à des situations nouvelles et d'apprendre en permanence. L'IA est directement issue des concepts cognitivistes, c'est à dire qu'elle envisage le fonctionnement cérébral sous un angle logico-déductif, et considère, de fait que l'acte cognitif s'effectue à travers une manipulation de symboles élémentaires. A travers le cognitivisme, l'IA établit une analogie entre le fonctionnement cérébral et celui de l'ordinateur [8]. Ce qui amène à dire que le but de l'IA est de construire des machines qui réalisent des choses qui requièrent de l'intelligence lorsqu'elles sont faites par des humains. Le terme d'IA est "né" officiellement durant la Dartmouth Conférence de l'été 1956 par Mc Carthy [9], mais était déjà dans les préoccupations des chercheurs depuis longtemps.

## II.3/ Techniques de Calcul intelligent

Le principe de base des méthodes de Calcul "Intelligent" consiste à considérer les êtres vivants comme modèles d'inspiration, le but étant de simuler à l'aide des machines leur comportement.

En général, ces techniques peuvent être regroupées en trois grandes classes : les réseaux de neurones artificiels qui utilisent l'apprentissage pour résoudre des problèmes complexes tels que la reconnaissance des formes ou le traitement du langage naturel, la logique floue utilisée dans des applications d'intelligence artificielle, dans lesquelles les variables ont des degrés de vérité représentés par une gamme de valeurs situées entre 1 (vrai) et 0 (faux), et les méthodes de calcul évolutif pour la recherche de l'optimisation (Figure II.1). Ces différentes classes seront présentées dans les sections suivantes :



**Figure II.1** : Différentes méthodes de l'intelligence computationnelle

## II.4/ L'algorithme génétique

Les Algorithmes Génétiques (en abrégé *AG*) sont des méthodes d'exploration de l'ensemble des solutions d'un problème utilisant les mêmes mécanismes que ceux intervenant dans la sélection naturelle. Ils sont utilisés principalement dans les domaines de l'optimisation et de l'apprentissage. Le parallélisme implicite et l'exploration globale de l'espace des solutions sont les deux principaux avantages des algorithmes génétiques. Tout d'abord, les algorithmes génétiques peuvent traiter des problèmes pour lesquels les solutions potentielles sont situées dans un espace de grande dimension, ce qui ne permet pas d'utiliser des méthodes standard reposant sur une exploration systématique. Ensuite, la recherche d'une solution optimale à un problème peut être réalisée par un algorithme génétique sans nécessiter de connaissance a priori sur la *répartition* des solutions dans l'espace[10] .

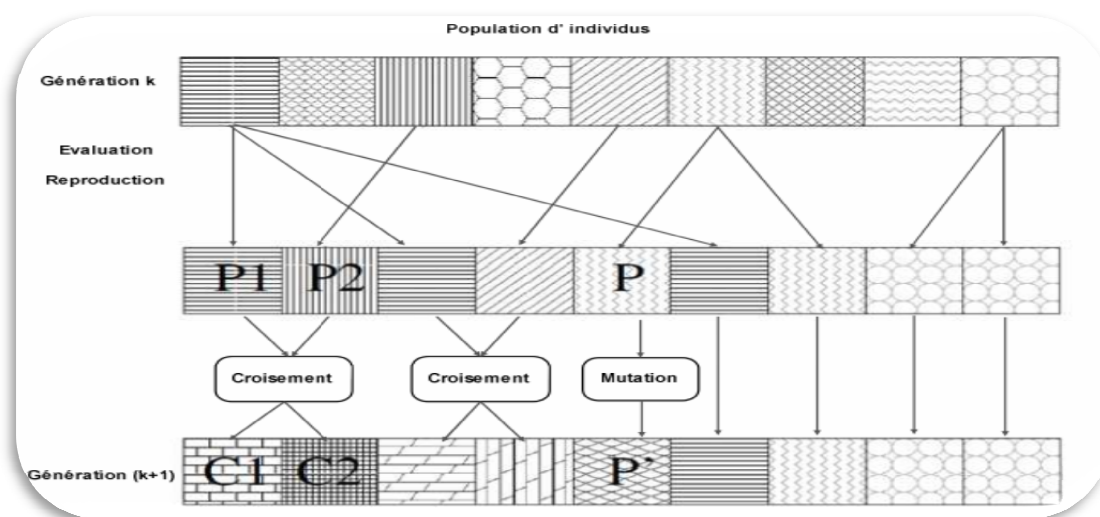
### II .4.1/ Concepts et principe des algorithmes génétiques :

Les algorithmes génétiques sont des algorithmes évolutionnistes d'optimisation stochastique inspirés des mécanismes de l'évolution naturelle (sélection, adaptation, reproduction, recombinaison, mutation) élaborés par Charles Darwin. Ils sont des algorithmes itératifs de recherche globale dont le but est d'optimiser une fonction prédéfinie appelée

critère ou fonction coût "fitness". Ils permettent de faire évoluer un ensemble initial de solutions vers un ensemble final. Les algorithmes génétiques sont utilisés dans l'automatique, le but de cette utilisation est d'optimiser au maximum les gains des régulateurs utilisés dans la commande. A partir des années 1990, quelques publications évoquent l'utilisation des AG pour la résolution de problèmes d'optimisation dans le domaine de la commande de systèmes électromécaniques.

Un algorithme génétique recherche le ou les extrema d'une fonction définie sur un espace de données. Pour l'utiliser, on doit disposer des cinq éléments suivants :

- Un principe de codage de l'élément de population.
- Un mécanisme de génération de la population initiale.
- Une fonction à optimiser appelée fitness ou fonction de l'évaluation des individus.
- Des opérateurs permettant de diversifier la population au cours des générations et d'explorer l'espace d'état. L'opérateur de croisement recompose les gènes d'individus existant dans la population, l'opérateur de mutation a pour but de garantir l'exploration de l'espace d'état.
- Des paramètres de dimensionnement: La taille de la population, nombre total de générations ou critère d'arrêt, probabilités d'application des opérateurs de croisement et de mutation [12].



**Figure II.2 :** Principe des algorithmes génétiques

Afin de trouver la solution optimale d'un problème en utilisant l'algorithme génétique, on commence par générer un ensemble de solutions (population d'individus) de façon aléatoire. L'évolution d'une génération à la suivante, utilise les trois opérations (Sélection, Croisement et Mutation) qui sont appliquées à tous les éléments de populations. Des couples de parents  $P_1$

et  $P_2$  sont sélectionnés en fonction de leurs adaptations. L'opérateur de croisement est appliqué avec une probabilité  $P_c$  et génère des couples d'enfants  $E_1$  et  $E_2$ . Ensuite, l'opérateur de mutation est appliqué aux enfants avec la probabilité  $P_m$  et génère des individus mutés  $E_1'$  et  $E_2'$  qui vont être insérés dans la nouvelle population. Parmi les critères d'arrêt qui peuvent être choisis pour l'algorithme, on peut citer :

- L'atteinte d'un nombre maximal de générations fixé a priori.
- L'obtention après un certain nombre de générations, d'un degré d'uniformité des individus de la population supérieur à un seuil souhaité.

#### II 4.1.1/La mise en œuvre d'un algorithme génétique :

La mise en œuvre d'un algorithme génétique est réalisée suivant les étapes suivantes :

- Création d'une population initiale.
- Evaluation des individus de la population.
- Sélection des meilleurs individus.
- Reproduction (Croisement et mutation).
- Formation d'une nouvelle génération.

La figure suivante montre l'organigramme de fonctionnement d'un algorithme génétique.

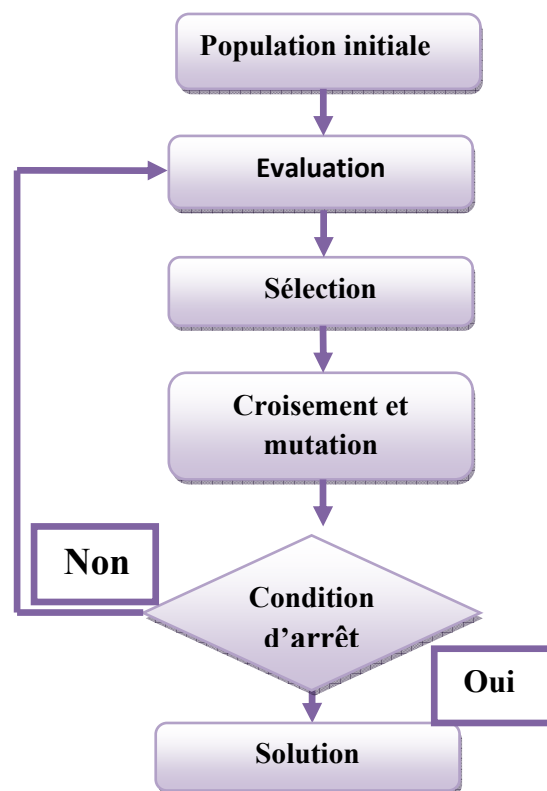


Figure II.3 : Organigramme de l'algorithme génétique

### II.4.1.2/Vocabulaires des algorithmes génétiques :

Dans cette section nous introduisons quelques vocabulaires utilisés dans la mise en œuvre des algorithmes génétiques.

- Individu : représenté par un chromosome (génome).
- Un chromosome est une chaîne de gènes.
- Génotype : l'ensemble des gènes représentés par un chromosome.
- Phénotype : l'ensemble des valeurs observables prises par chaque gène.
- Fonction d'adaptation : fitness.
- Opération de reproduction : Le croisement, la mutation.
- Génération : l'ensemble de la population à un moment donné du processus.

### II.4.2/ Le codage :

Le codage est une modélisation d'une solution d'un problème donné sous forme d'une séquence de caractères appelée chromosome où chaque caractère, dit aussi gène, représente une variable ou une partie du problème. La tâche principale consiste à choisir le contenu des gènes qui facilite la description du problème et respecte ses contraintes. Il utilise principalement deux types de codage : le codage binaire, le codage réel :

#### *a) Codage binaire :*

Dans ce type de codage le chromosome représente simplement une suite de 0 et de 1. Le codage binaire est également indépendant des opérateurs génétiques (croisement et mutation) du moment où ces derniers ne nécessitent aucune spécification. En effet, toute manipulation d'un chromosome donne naissance à un nouveau chromosome valide. Cependant, ce type de codage n'est pas toujours bon, il est parfois très difficile ou très lourd de coder des solutions de cette manière. En outre, dans certain cas la taille mémoire requise peut devenir prohibitive.

#### *b) Codage réel :*

Dans ce type de codage, chaque chromosome est un vecteur dont les composantes sont les paramètres du processus d'optimisation. Par exemple, si on recherche l'optimum d'une fonction de  $n$  variables  $f(x_1, x_2, \dots, x_{n-1}, x_n)$ , on peut utiliser tout simplement un chromosome contenant les  $n$  variables :  $(x_1, x_2, \dots, x_{n-1}, x_n)$ , avec ce type de codage, la procédure d'évaluation des chromosomes est plus rapide vu l'absence des étapes de codage et de transcodage (du réel vers le binaire et vice-versa).

### **II.4.3/ La population initiale**

Le choix de la population initiale d'individus conditionne fortement la rapidité de l'algorithme. Pour générer la population initiale, nous avons, deux possibilités. Dans le cas où aucune information sur la position de la solution n'est disponible, le but est de recouvrir au mieux l'espace d'état. Une génération aléatoire est donc engendrée par des tirages uniformes sur chaque gène du chromosome. Dans le cas où l'utilisateur connaîtrait un sous-domaine de l'espace d'état où la solution se trouve, la population initiale est générée dans ce sous-espace.

Dans le cas d'une génération engendrée de façon aléatoire, il existe deux éventualités. Soit il est possible de savoir à l'avance si un point respecte les contraintes du problème d'optimisation, la population est alors générée aléatoirement dans le domaine admissible. Soit il n'est pas possible de le savoir, le respect des contraintes sera alors assuré via l'ajout d'une pénalisation dans la fonction- objectif et la population est uniformément distribuée sur tout l'espace d'état.

### **II.4.4/ Fonction d'évaluation (fitness)**

L'évaluation de l'adaptation de chaque individu à l'environnement est réalisée au moyen d'une fonction d'adaptation (fitness). Cette fonction attribue à chaque individu une valeur qui représente son niveau d'adaptation. La fonction d'adaptation peut affecter directement la qualité des résultats obtenus par l'AG ainsi que le temps d'exécution. Elle est fonction du critère à minimiser. En raison de l'analogie avec la théorie de l'évolution (survie des individus les mieux adaptés à leur environnement), l'algorithme génétique est naturellement formulé en terme de maximisation.

### **II.4.5/ Les opérateurs de base d'un AG**

Les opérateurs sont fondamentaux pour implanter le processus reproductif caractéristique d'un AG. De nombreux opérateurs génétiques existent ainsi que différentes stratégies d'implantation. Par contre, dans ce paragraphe nous n'abordons que les opérateurs les plus simples et les plus utilisés à savoir : l'opérateur de sélection, de croisement et de mutation, pour ne pas complexifier la méthode et perdre le sens physique des paramètres de réglage.

#### **II.4.5.1/ Le croisement**

La naissance d'un nouvel individu, nécessite la prise aléatoire d'une partie des gènes de chacun des deux parents. Ce phénomène, issu de la nature est appelé croisement (crossover). Il s'agit d'un processus essentiel pour explorer l'espace des solutions possibles. Une fois la sélection terminée, les individus sont aléatoirement répartis en couples. Les chromosomes parents sont alors copiés et recombinaison afin de produire chacun deux descendants ayant des

caractéristiques issues des deux parents. Dans le but de garder quelques individus parents dans la prochaine population, on associe à l'algorithme génétique une probabilité de croisement, qui permet de décider si les parents seront croisés entre eux ou s'ils seront tout simplement copiés dans la population suivante. Il existe plusieurs types de croisement parmi lesquels on trouve : le croisement en 1 point, le croisement en deux points et le croisement en N points, ces types sont résumés dans la figure suivante :

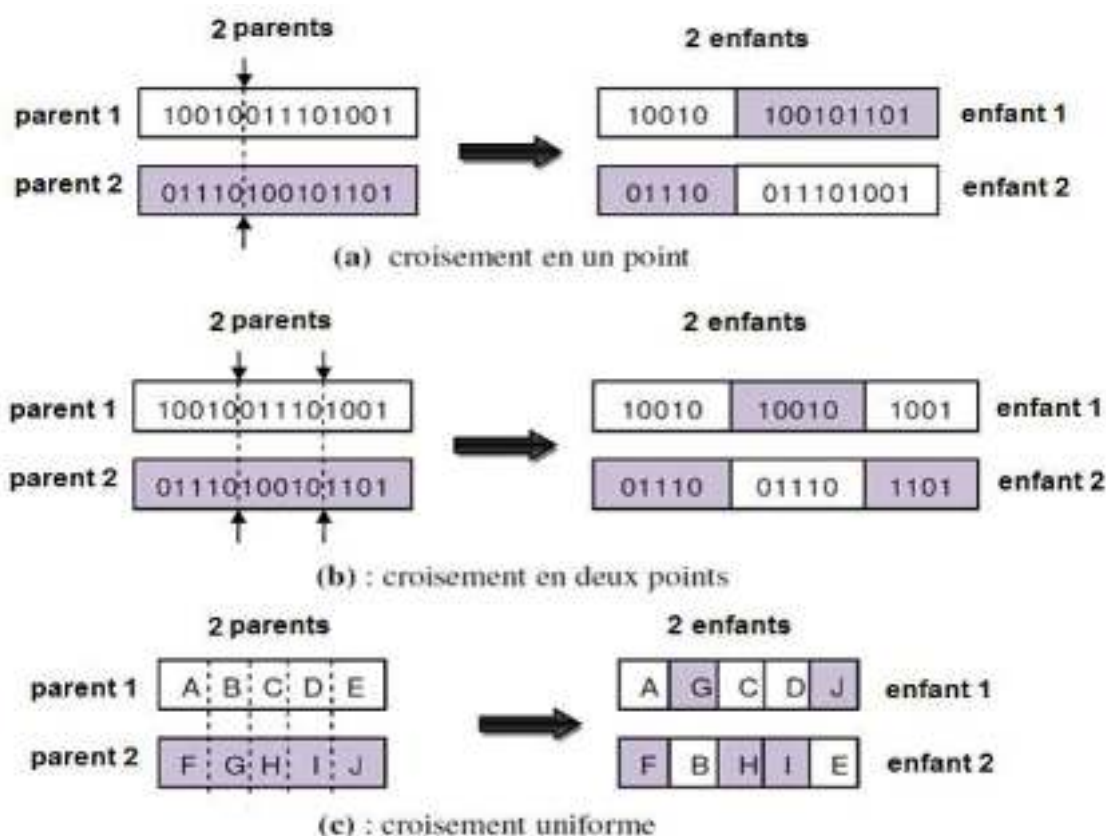


Figure II.4 : Exemples d'opération de croisement

#### II.4. 5.2/ La mutation

La mutation est un opérateur qui agit sur la population des enfants, son rôle est de garantir une diversité dans la population générée en introduisant de nouveaux gènes au patrimoine génétique de la population et ce avec une probabilité ( $P_m$ ) souvent très faible. L'opérateur de mutation diffère suivant le type du codage utilisé, ainsi dans le codage binaire la mutation est très simple à élaborer, il suffit de compléter à 1 toutes les allèles des chromosomes des enfants pour qui, la probabilité est inférieure à ( $P_m$ ). Figure II.4 donne un exemple illustratif pour la mise en œuvre de cet opérateur

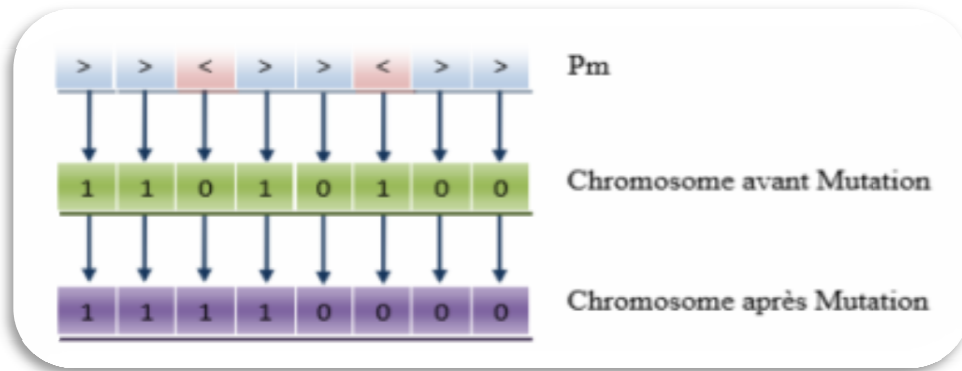


Figure II.5 : Principe de la mutation binaire

L'opérateur de mutation modifie donc de manière complètement aléatoire, les caractéristiques d'une solution, ce qui permet d'introduire et de maintenir la diversité au sein de notre population de solutions. Cet opérateur joue le rôle d'un "élément perturbateur", il introduit du "bruit" au sein de la population.

#### II.4.6/ La sélection

L'opérateur de sélection est chargé de "favoriser" les meilleurs individus. Plus formellement, l'opérateur de sélection va générer à partir de la population courante une nouvelle population par copie des individus choisis de la population courante. La copie des chaînes s'effectue en fonction des valeurs de la fonction d'adaptation. Ce procédé permet de donner aux meilleures chaînes, une probabilité élevée de contribuer à la génération suivante. Cet opérateur est bien entendu une version artificielle de la sélection naturelle, la survie darwinienne des chaînes les plus adaptées. Plusieurs stratégies sont possibles pour effectuer une telle sélection parmi lesquelles nous abordons :

✓ **La sélection par classement** : elle consiste à ranger les individus de la population dans un ordre croissant (ou décroissant selon l'objectif) et à retenir un nombre fixé de génotypes. Ainsi, seuls les individus les plus forts sont conservés. L'inconvénient majeur de cette méthode est la convergence prématurée de l'algorithme génétique.

✓ **La sélection par tournoi** : le tournoi le plus simple consiste à choisir aléatoirement un nombre  $k$  d'individus dans la population et à sélectionner celui qui a la meilleure performance. Les individus qui participent à un tournoi sont remis ou sont retirés de la population, selon le choix de l'utilisateur. Avec le tournoi binaire, sur deux individus en compétition, le meilleur gagne avec une probabilité  $p \in [0, 5; 1]$

### II.4.7/ La convergence d'un AG :

Le cycle de génération et de sélection de population est répété jusqu'à ce qu'un critère d'arrêt soit satisfait ; ce critère peut être notamment un nombre maximum de générations, un temps maximal de calcul, une valeur de fitness minimale, ou/et une convergence vers une solution satisfaisante. Les valeurs de tels paramètres dépendent fortement de la problématique étudiée. Ainsi il n'existe pas de paramètres qui soient adaptés à la résolution de tous les problèmes qui peuvent être posés à un algorithme génétique. Cependant, certaines valeurs sont souvent utilisées (définies dans la littérature) et peuvent être de bons points de départ pour démarrer une recherche de solutions à l'aide d'un AG.

- ✓ La probabilité de croisement est choisie dans l'intervalle [0.7, 0.99].
- ✓ La probabilité de mutation est choisie dans l'intervalle [0.001, 0.01].
- ✓ Taille de la population : entre N=30 et 50 individus

### II.4.8 / Développement d'un algorithme génétique pour le réglage PI

Pour faciliter le développement de ce projet, un système a été choisi et un contrôleur PI a été conçu à l'aide de méthodes conventionnelles. Un algorithme génétique a ensuite été créé pour évaluer les coefficients PI du même système et les résultats des deux techniques ont été comparés.

#### II.4.8.1/ Écrire la fonction objective pour un algorithme génétique :

Écrire une fonction objective est la partie la plus difficile de la création d'un algorithme génétique. Dans ce projet, la fonction objective est nécessaire pour évaluer le meilleur contrôleur PI pour la MCC. Une fonction objective peut être créée pour trouver un contrôleur PI offrant le plus petit dépassement, le temps de montée le plus rapide ou le temps de réglage le plus rapide. Toutefois, afin de combiner tous ces objectifs, il a été décidé de concevoir une fonction objective qui minimise l'erreur du système contrôlé.

Le contrôleur PI est placé dans une boucle de retour unité avec la fonction de transfert. Afin de réduire le temps de compilation du programme, la fonction de transfert est définie dans un autre fichier et importée en tant que variable globale. Le système contrôlé reçoit une entrée pas à pas et l'erreur est évaluée en utilisant un critère de performance d'erreur approprié, à savoir ITAE, ISE, IAE ou MSE.

### II.4.8.2/ Indices de performance

Diverses fonctions objectives ont été écrites sur la base de critères de performance d'erreur. Chaque fonction d'objective est fondamentalement la même, à l'exception de la section du code qui définit le critère de performance d'erreur spécifique mis en œuvre. Pour optimiser les performances d'un système contrôlé par PI, les gains PI du système sont ajustés pour maximiser ou minimiser un certain indice de performance. L'indice de performance est calculé sur un intervalle de temps T, normalement dans la région de  $0 \leq t \leq t_0$  où  $t_0$  est le temps d'établissement du système. Les indices de performance utilisés étaient:

- Intégrale du temps multipliée par l'erreur absolue (ITAE) :

$$I_{ITAE} = \int_0^T t|e(t)|dt \quad (\text{II.1})$$

- Intégrale de la grandeur absolue de l'erreur (IAE) :

$$I_{IAE} = \int_0^T |e(t)|dt \quad (\text{II.2})$$

- Intégrale du carré de l'erreur (ISE) :

$$I_{ISE} = \int_0^T (e(t))^2 dt \quad (\text{II.3})$$

- Moyenne du carré de l'erreur (MSE) :

$$I_{MSE} = \frac{1}{n} \sum_{i=0}^n (e(t))^2 \quad (\text{II.4})$$

### II.4.9/ Application des AG à l'optimisation du régulateur PI pour la commande d'une machine à courant continu:

Dans cette partie nous présentons l'optimisation des paramètres du PI par l'utilisation des AG. L'objectif est toujours de trouver l'ensemble des paramètres optimisés PI de façon à ce que la réponse en vitesse et en courant du système en boucle fermée soit stable et plus robuste.

#### II.4.9.1/ Boucle interne de régulation du courant :

Le régulateur de courant direct permet de définir la tension V, on aura :

$$FT_{\text{elect}} = \frac{1}{L.P+R} \quad (\text{II.5})$$

La boucle de régulation du courant peut se représenter par le schéma bloc suivant :

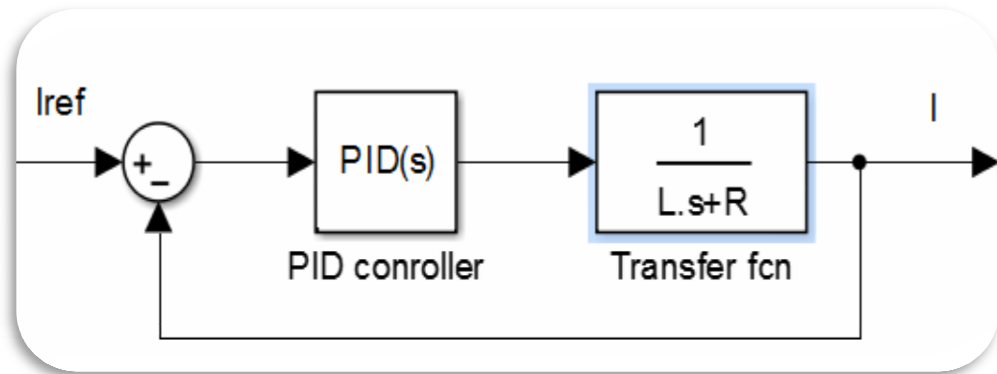


Figure II.6 : Schéma bloc de la boucle de courant

L'algorithme génétique est compilé à l'aide de la commande. Une fois cette commande est entrée, et à partir de l'interface GATOOOL, l'algorithme génétique effectue une itération jusqu'à ce qu'il réponde aux critères décrits par sa fonction de terminaison.

L'interface GATOOOL :

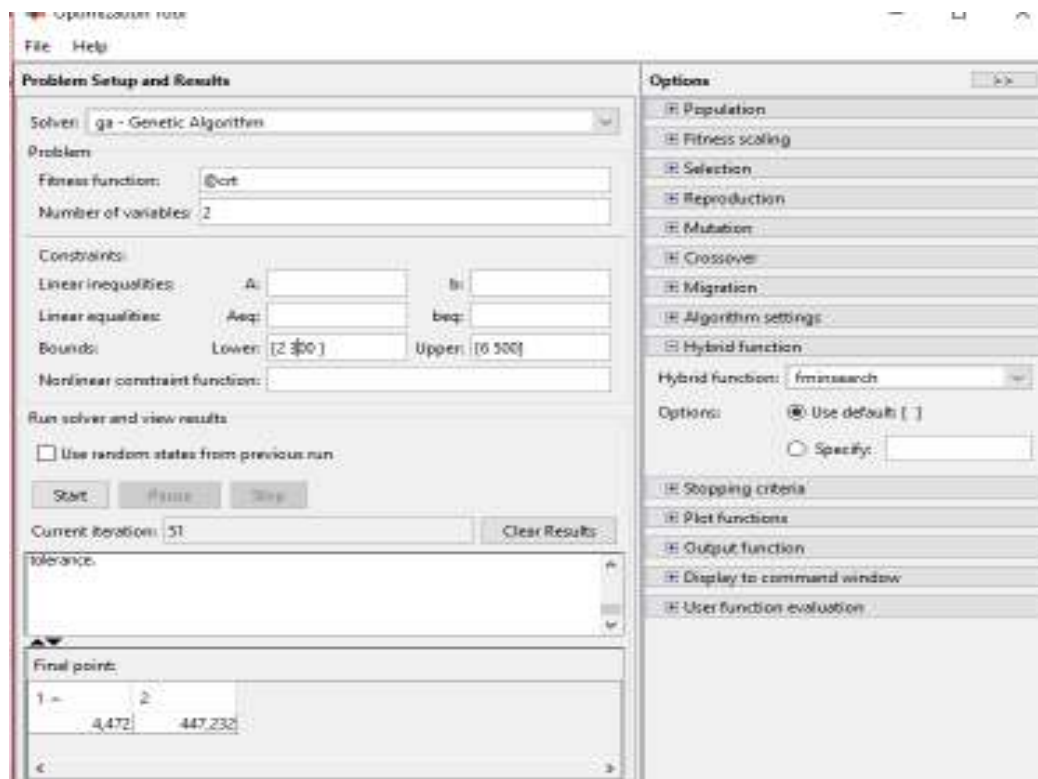


Figure II.7 : Interface GATOOOL pour l'assertivement du courant

- **Fitness fonction** : est la fonction objective que nous souhaitons minimiser. Nous pouvons spécifier la fonction comme un descripteur de fonction de la forme @crt, où crt.m est un fichier M qui renvoie un scalaire.

- **Nombre de variables** : est le nombre de variables indépendantes pour la fonction fitness.
- **Bornes** : sont les limites inférieure et supérieure pour les variables:
  - Lower (Inférieur) = spécifie les limites inférieures en tant que vecteur.
  - Upper (Supérieur) = spécifie les limites supérieures en tant que vecteur.

Une fois l'algorithme génétique est terminé, il renvoie les variables  $K_p$  et  $K_i$  Nous avons effectué plusieurs expériences pour déterminer ces paramètres :

➤  $k_p = 4.47$  et  $k_i = 447.23$

➤  $PI = 4.47 + \frac{447.25}{P}$

La réponse du système en vitesse avec ces paramètres :

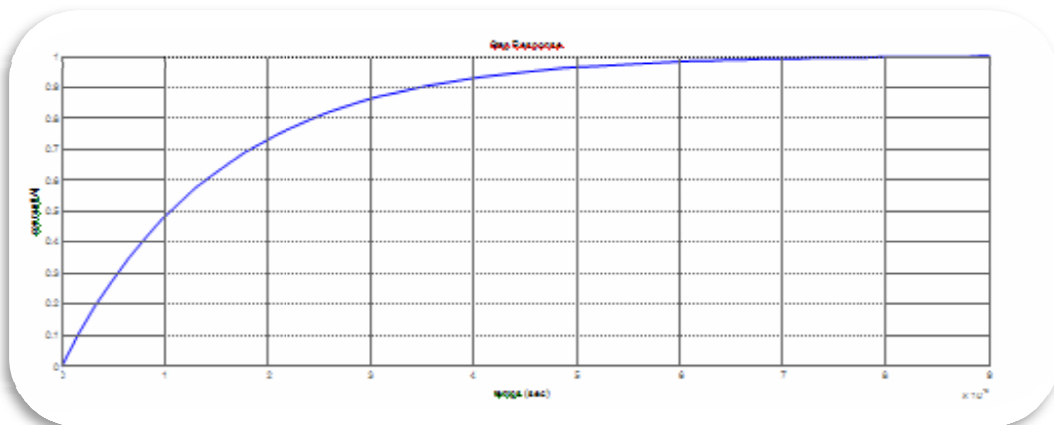


Figure II.8 : Réponse en courant à partir des paramètres du régulateur  $PI_{AG}$

#### II.4.9.2/ Boucle externe de régulation de la vitesse:

Le régulateur de vitesse permet de déterminer le courant de référence; soit l'équation mécanique donnée :

$$FT_{mec} = \frac{1}{JP+f} \quad (II.6)$$

La boucle de régulation de la vitesse peut se représenter par le schéma bloc suivant :

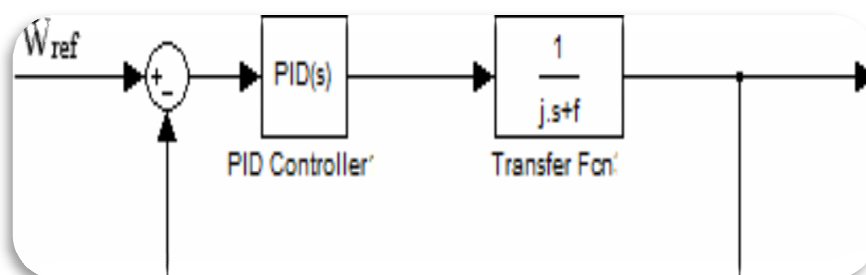
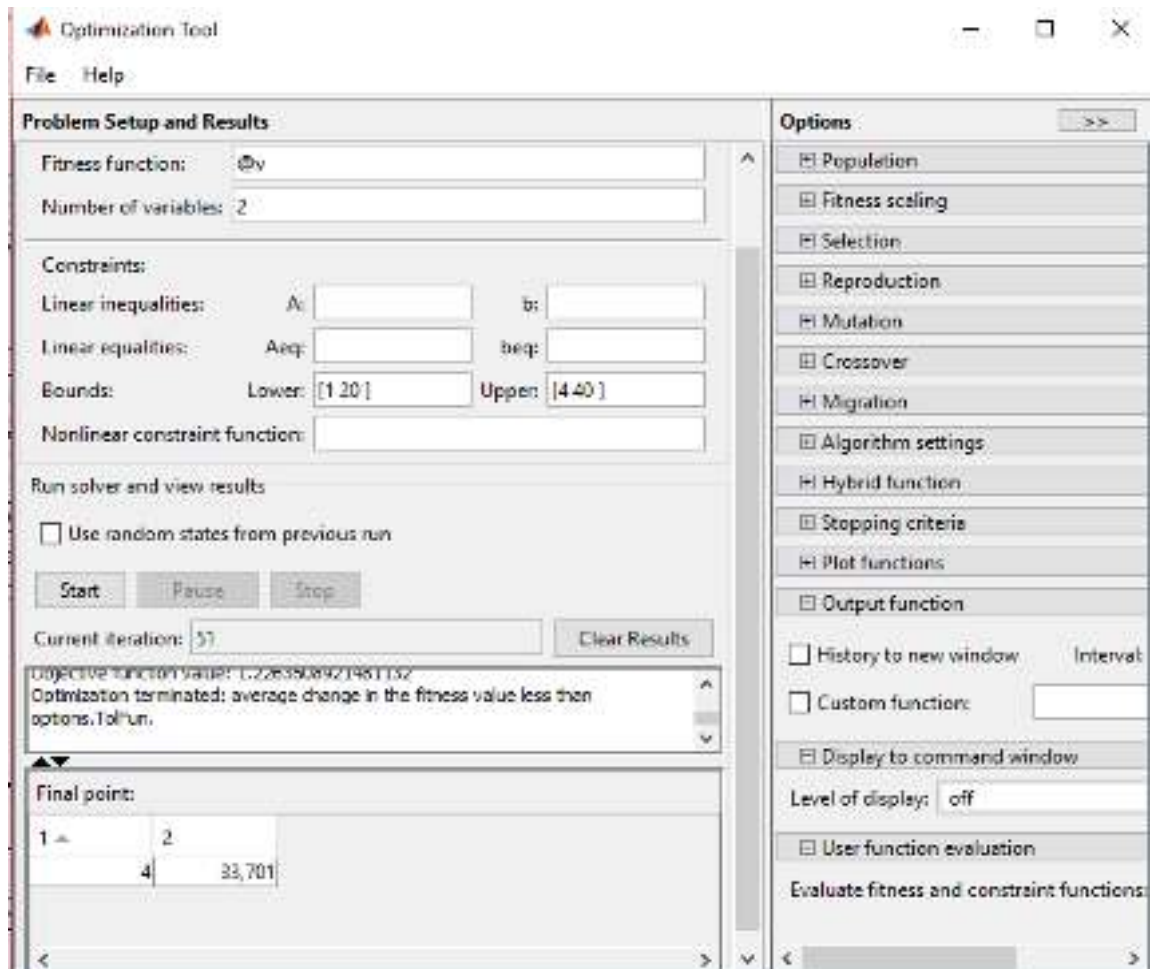


Figure II.9 : Schéma bloc de la boucle de vitesse

L'algorithme génétique est compilé à l'aide de la commande. Une fois cette commande est entrée, et à partir de l'interface GATool, l'algorithme génétique effectuera une itération jusqu'à ce qu'il réponde aux critères décrits par sa fonction de terminaison.

A partir de l'interface GATool :



**Figure II.10:** Interface GATool pour l'asservissement de la vitesse

Une fois l'algorithme génétique est terminé, il renvoie les variables  $K_p$  et  $K_i$

Nous avons effectué plusieurs expériences pour déterminer ces paramètres.

La réponse du système en vitesse avec ces paramètres :

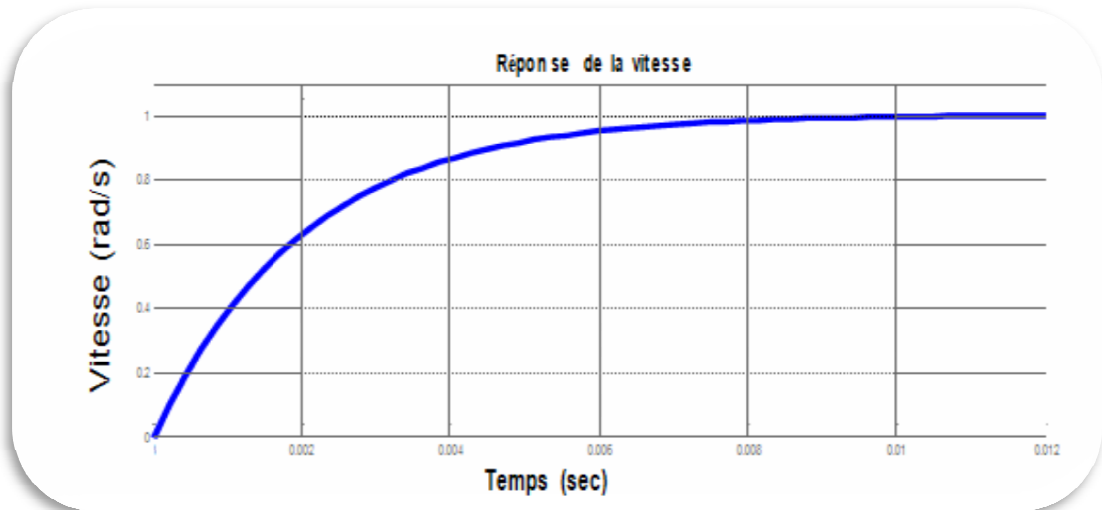


Figure II.11: Réponse en vitesse à partir des paramètres du régulateur PI<sub>AG</sub>

**II.4.9.3/ Application des paramètres du régulateur PI optimisés sur le modèle de la MCC :**

➤ Après avoir obtenu les différentes valeurs du régulateur optimisées, on a relevé les valeurs des paramètres :  $k_p=4.47$  et  $k_i=447$  pour l’asservissement du courant et :  $k_p=4$  et  $k_i=33.7$  pour l’asservissement de la vitesse, on applique ces résultats sur la machine à courant continu.

**❖ Résultats de simulation**

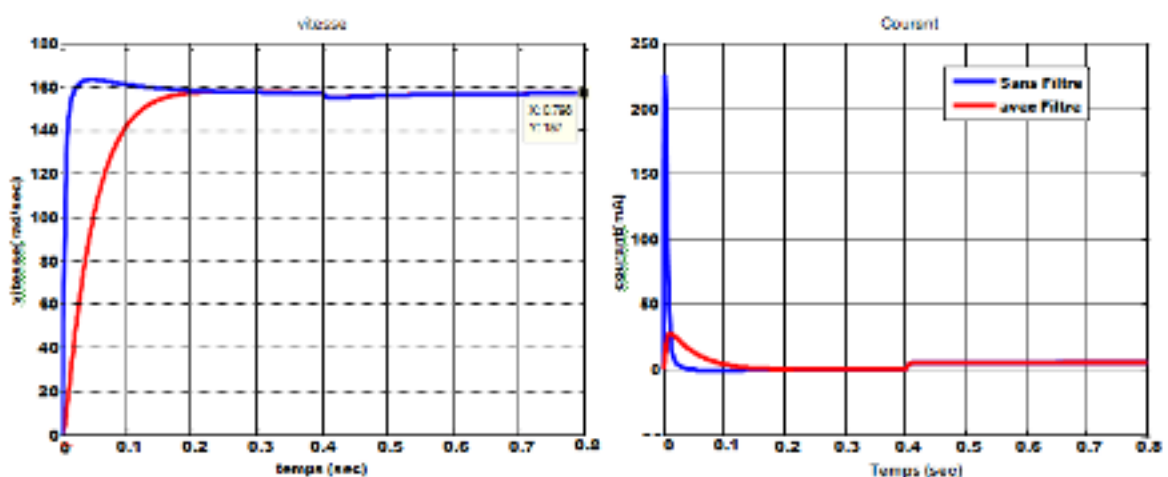


Figure II.12 : Réponses en vitesse et en courant optimisées par l’AG

### ❖ **Interprétation :**

**Vitesse :** L'allure de la vitesse sans filtre possède une caractéristique presque linéaire et atteint la vitesse de référence dans un temps de réponse  $t_r=0.02$  sec et un temps de montée  $t_m=0.01$  sec avec un dépassement de 8%. Après l'application de la charge à l'instant  $t=0.3$  sec de 5Nm, on ne constate presque aucune influence sur l'allure de la vitesse, et L'allure de la vitesse avec filtre possède une caractéristique exponentielle et atteint la vitesse de référence dans un temps de réponse  $t_r=0.15$  sec et un temps de montée  $t_m=0.1$  sec. Avec un dépassement de 0%. Après l'application de la charge à l'instant  $t=0.3$  sec de 5Nm, on ne constate presque aucune influence sur l'allure de la vitesse par le régulateur PI.

**Le courant :** L'allure de courant sans filtre subit au moment du démarrage un grand pic de 400A contrairement l'allure de courant avec filtre subit un pic de 39A puis atteint rapidement la valeur nulle, au moment d'application de la charge, le courant a augmenté pour équilibrer la charge (couple résistant),

### **II.4.9.4/ conclusion**

Nous pouvons conclure que les AG sont des algorithmes simples de conception et peuvent résoudre des problèmes assez complexes. La résolution de ces problèmes est obtenue grâce aux opérateurs de reproduction. Ces AG sont des procédures assez robustes pour résoudre un problème d'optimisation pour la sélection des primitives. Nous avons observé que certains résultats obtenus sont mauvais à cause d'une mauvaise initialisation du domaine de recherche ou par un mauvais choix de paramètres. Lors de l'initialisation du domaine de recherche, il est difficile d'atteindre la solution. Pour déterminer des bons paramètres et un bon choix des méthodes de reproduction, il est nécessaire d'effectuer plusieurs expériences

### **II.5/ La logique floue**

La logique floue constitue aujourd'hui une approche alternative intéressante. Elle présente plusieurs avantages tels que le raisonnement proche de celui de l'homme, sa capacité à contrôler des performances dynamiques et ses qualités intéressantes de robustesse. La logique floue inspire ses caractéristiques du raisonnement humain. Elle est basée sur la constatation que la plupart des phénomènes ne peuvent être représentés à l'aide de variables booléennes qui ne peuvent prendre que deux valeurs 0 ou 1. La logique floue est donc un outil pour la

manipulation et la représentation des connaissances imparfaitement décrites, ses objets sont modélisés par des ensembles flous support de base des notions vagues.

Enfin, elle présente aussi l'intérêt d'être plus facile à implémenter qu'une logique probabiliste.

## **II.5.1/ Historique de la logique floue**

### **II.5.1.1/ Apparition de la logique floue**

Le terme d'ensemble flou apparaît pour la première fois en 1965 lorsque le professeur Lotfi Zadeh, de l'université de Berkeley aux USA, publie un article intitulé « Ensembles flous » (Fuzzy sets). Il a réalisé depuis de nombreuses avancées théoriques majeures dans le domaine et a été rapidement accompagné par de nombreux chercheurs développant des travaux théoriques.

### **II.5.1.2/ Premières applications**

Parallèlement, certains chercheurs se sont penchés sur la résolution par logique floue de problèmes réputés difficiles. Ainsi en 1975, le professeur Mamdani à Londres développe une stratégie pour le contrôle des procédés et présente les résultats très encourageants qu'il a obtenus sur la conduite d'un moteur à vapeur. En 1978, la société danoise F.L.Smith réalise le contrôle d'un four à ciment. C'est là la première véritable application industrielle de la logique floue.

### **II.5.1.3/Développement**

C'est au Japon où la recherche n'est pas seulement théorique mais également très applicative, que la logique floue connaît son véritable essor. Dans l'industrie, le traitement des eaux, les grues portuaires, les métros, les systèmes de ventilation et de climatisation sont touchés. Enfin, des applications existent dans des domaines très différents tels que la finance ou le diagnostic médical. A partir de 1990, c'est en Allemagne que des applications apparaissent en grand nombre ainsi qu'à une moindre échelle aux USA. Aujourd'hui, une vaste gamme de nouveaux produits ont une étiquette « produit flou (Fuzzy) ».

## **II.5.2/ Avantages et inconvénients de la logique floue**

Évidemment, le réglage par la logique floue réunit un certain nombre d'avantages et inconvénients. Les avantages essentiels sont :

- La théorie est simple et s'applique à des systèmes complexes.
- Robustesse de la commande floue vis-à-vis des incertitudes.
- Possibilités de commande auto-adaptative aux variations du procédé.

- La maîtrise du système à régler avec un comportement complexe.
- L'obtention fréquente de meilleures prestations dynamiques (régulateur non linéaire).

Par contre les inconvénients sont :

- Technique de réglage essentiellement empirique.
- Performances dépendent de l'expertise.
- Le manque de directives précises pour la conception d'un réglage (choix de grandeurs à mesurer, détermination de la fuzzification, des inférences et de la défuzzification).
- L'approche artisanale et non systématique (implémentation des connaissances de l'opérateur, est souvent difficile)

### **II.5.3/ Pourquoi utiliser la logique floue ?**

Il convient d'utiliser la logique floue lorsque des imperfections entachent la connaissance dont nous disposons sur le système, lorsqu'une modélisation rigoureuse de celui-ci est difficile, lorsqu'il est très complexe, lorsque notre façon naturelle de l'aborder passe par une vue globale de certains de ses aspects.

Si elle impose une standardisation de la signification des descriptions du système exprimées linguistiquement, ce qui peut sembler rédacteur par rapport à une utilisation purement symbolique des descriptions linguistiques, la logique floue présente l'avantage de permettre le passage d'une telle description d'un expert ou d'un observateur à l'autre

### **II.5.4/Théorie de sous ensemble flou**

#### **II.5.4.1/Concept de sous-ensemble flou**

Le concept de sous-ensemble flou a été introduit pour éviter les passages brusques d'une classe à une autre (dans la classe noir à la classe blanche par exemple) et autoriser des éléments à n'appartenir complètement ni à l'une ni à l'autre, ou encore à appartenir partiellement à chacune (avec un fort degré à la classe noire et un faible degré à la classe blanche dans le cas du degré foncé).

La notion de sous-ensemble flou permet de traiter :

- Des catégories aux limites mal définies (comme "centre ville "ou "ancien").
- Des situations intermédiaires entre le tout et le rien ("presque noir").

Le concept de sous-ensemble flou constitue un assouplissement de celui de sous-ensemble flou d'un ensemble donné, il n'existe pas d'ensemble flou sens propre, tous les ensembles considérés étant classique et bien définis.

### II.5.4.2/Définition d'un sous-ensemble flou

Les sous-ensembles flous constituent une extension des ensembles classiques. Contrairement à l'algèbre booléenne qui associe à chacun des éléments la valeur 0 ou 1, La théorie des sous-ensembles flous attribue un degré de confiance compris dans l'intervalle  $[0, 1]$  à chacun de ses éléments. De par cette représentation, cette approche, qualifiée d'ensablisme, permet donc de modéliser une imprécision sur la connaissance des bornes d'un paramètre.

Un sous-ensemble flou  $A$  d'un référentiel  $X$  est caractérisé par une fonction d'appartenance  $\mu_A$  :

Si  $\mu_A$  est la fonction d'appartenance de l'ensemble flou  $A$

$$\forall x \in X, \mu_A \in [0,1]$$

L'ensemble  $A$  est défini par :  $A = \{(X, \mu(x)) / x \in X\}$

### II.5.4.3/ Fonction d'appartenance

Au lieu d'appartenir à l'ensemble « vrai » ou à l'ensemble « faux » de la logique binaire traditionnelle, la logique floue admet des degrés d'appartenance à un ensemble donné. Le degré d'appartenance à un ensemble flou est matérialisé par un nombre compris entre 0 et 1. Une valeur précise de la fonction d'appartenance liée à une valeur de la variable est notée  $\mu$  et appelée « facteur d'appartenance ». La fonction d'appartenance décrivant un sous-ensemble flou est caractérisée par quatre propriétés :

- **Le type** : La forme du nombre flou qui peut être triangulaire, trapézoïdale, gaussienne ou sigmoïdale.
- **La hauteur** :  $H(A) = \text{Sup}_{x \in X} (\mu_A(x))$  de la fonction d'appartenance, un sous ensemble flou est dit normalisé s'il est de hauteur 1.
- **Le noyau** :  $N(A) = \{x | \mu_A(x) = 1\}$  est l'ensemble des éléments qui appartiennent totalement à  $A$ , pour les fonctions de type triangulaire, le noyau est un singleton qui est appelé aussi valeur modale.
- **Le support** :  $S(A) = \{x | \mu_A(x) \neq 0\}$ , cet ensemble décrit l'ensemble des éléments qui sont partiellement dans  $A$ .

🧩 Le type

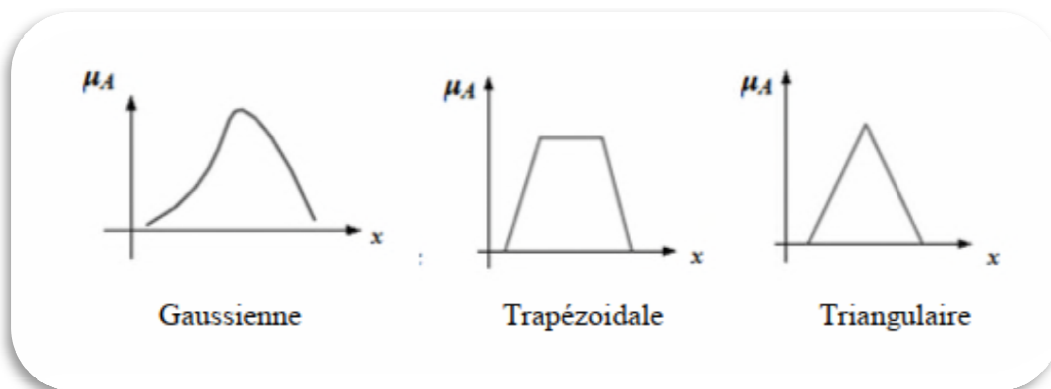


Figure II.13: Représentations des différents types

🧩 La hauteur, Le noyau, Le support

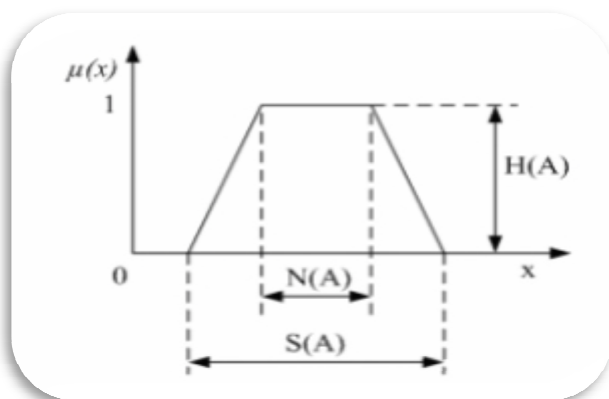


Figure II.14 : Représentations de La hauteur, Le noyau, Le support

II.5.4.4/ Opérations sur les sous ensembles flous

Supposons que A et B sont deux sous-ensembles flous définis dans un univers du discours X par les fonctions d'appartenance  $\mu_A$  et  $\mu_B$ . [3]

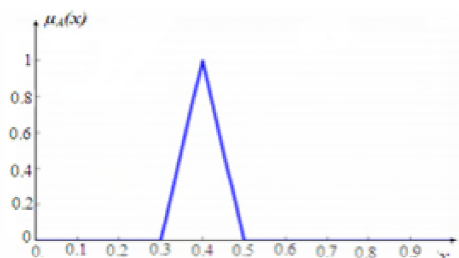


Figure II.15: Fonction d'appartenance A

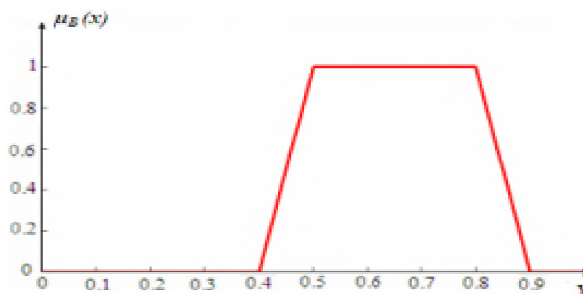


Figure II.16: Fonction d'appartenance B

On peut définir des opérations ensemblistes telles que l'égalité, l'inclusion, l'intersection, l'union et le complément grâce à des opérations sur les fonctions d'appartenance.

- **Égalité** : A et B sont dits égaux, propriété que l'on note  $A = B$ .

Si leurs fonctions d'appartenance prennent la même valeur en tout point de X :

$$\forall x \in X \quad \mu_A(x) = \mu_B(x)$$

- **Inclusion** : A est dit inclus dans B, propriété que l'on note  $A \subset B$ ,

Si tout élément x de X qui appartient :

$$\forall x \in X \quad \mu_A(x) \leq \mu_B(x)$$

- **La conjonction floue 'ET' ou Intersection** : L'intersection de A et B, que l'on note  $A \cap B$ , est le sous-ensemble flou  $\mu$  constitué  $\mu$  des éléments de X affectés du plus petit des deux degrés d'appartenance  $\mu_A(x)$  et  $\mu_B(x)$

$$\forall x \in X \quad \mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$$

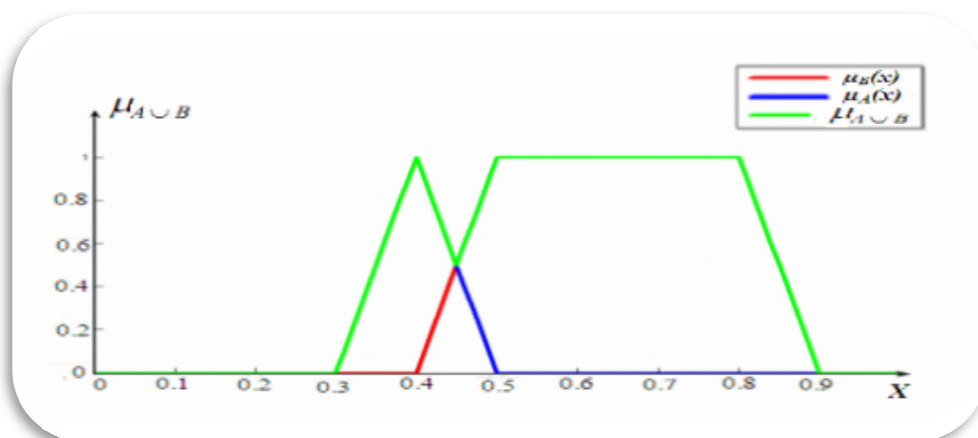
- **La disjonction floue 'OU' ou Union** : L'union de A et B, que l'on note  $A \cup B$ , est le sous-ensemble flou constitué des éléments de X affectés du plus grand des deux degrés d'appartenance  $\mu_A(x)$  et  $\mu_B(x)$  :

$$\forall x \in X \quad \mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$$

- **La Complémentation, négation ou inverse** : Le complément de A, que l'on note  $A^c$ , est le sous-ensemble flou de X constitué des éléments x lui appartenant d'autant plus qu'ils appartiennent peu à A :

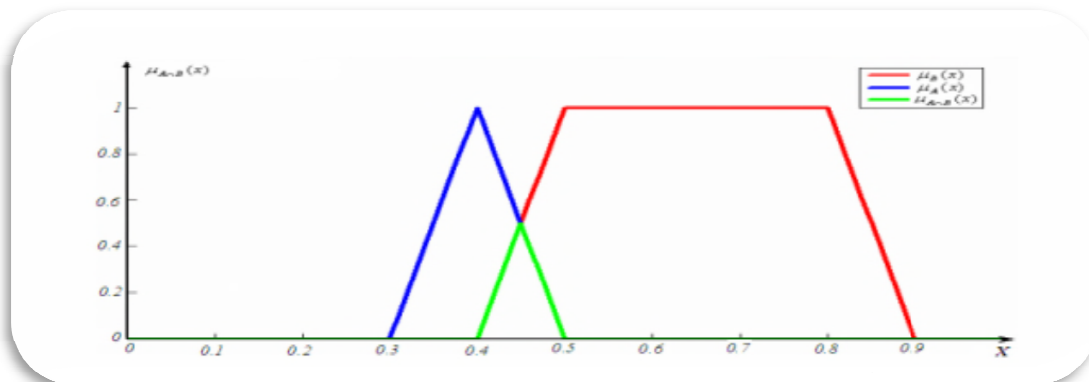
$$\forall A \in X \quad \mu_{(A^c)}(x) = 1 - \mu_A(x)$$

La fonction d'appartenance (**Figure II-17**) « $\mu_A \cup \mu_B$  » est définie, pour tout  $u \in U$ , par :



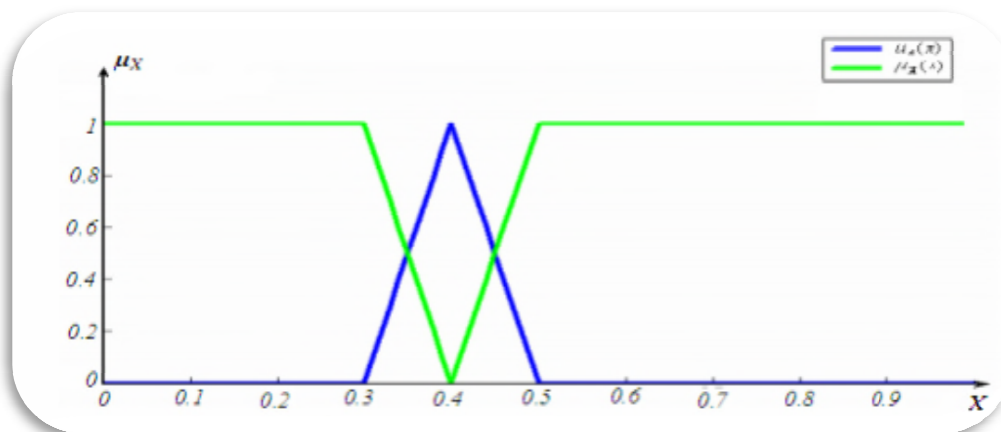
**Figure II.17:** Fonction d'appartenance de l'union

La fonction d'appartenance (**Figure II-18**) « $\mu_A \cap \mu_B$  » est définie, pour tout  $u \in U$ , par :



**Figure II.18 :** Fonction d'appartenance d'intersection

La fonction d'appartenance (**Figure II-19**) « $\mu_A$ » du complément d'un ensemble A est définie, pour tout  $u \in U$ , dans les deux cas par :



**Figure II.19:** Opération de complémentation de la fonction d'appartenance A

### II.5.5/ Commande par logique floue

Bien que la logique floue possède un champ d'application extrêmement vaste, nous nous intéressons ici qu'à son utilisation dans le cadre de la commande. En effet, cette méthode permet d'obtenir une loi de commande souvent très efficace sans devoir faire des études théoriques approfondies. L'algorithme de commande consiste en une collection de règles

floues appelées base de règle. Dans cette section nous présentons toutes les bases générales du réglage et de la commande par logique floue.

### II.5.5.1/ Structure d'un régulateur flou

Le régulateur par logique flou est composé de quatre éléments de base suivants :

- Interface de fuzzification
- Base de connaissances
- Moteur d'inférence floue
- Interface de défuzzification

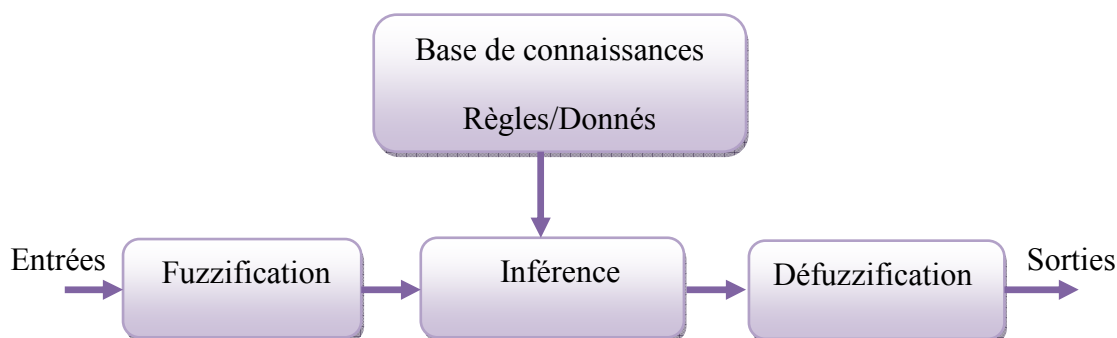


Figure II.20 : Structure de base d'un régulateur flou

#### II.5.5.1.1/Base de connaissance

Elle contient les définitions des fonctions d'appartenance (formes et paramètres) associées aux variables d'entrée/sortie ainsi que l'ensemble des règles floues. Le problème initial est alors décomposé en un ensemble de règles, de la forme :

“SI variable EST propriété, ALORS action”

Qui définit la réponse désirée du système en sortie pour les conditions d'entrée du système.

Le nombre et la complexité des règles dépendent du nombre de paramètres d'entrée qui doivent être utilisés et du nombre de variables floues associées à chaque paramètre.

Dans le cas où plusieurs données d'entrée sont retenues, les règles vont s'exprimer par exemple comme suit :

Si  $x_1$  est A1 ET  $x_2$  est A2 alors  $y$  est B

Si  $x_1$  est A1 OU  $x_2$  est A2 alors  $y$  est B

Où  $x_1$ ,  $x_2$  et  $y$  sont les grandeurs physiques caractéristiques du système, A1, A2 et B sont les termes linguistiques, « OU » et « ET » les opérateurs de liaison des fonctions d'entrée.

### II.5.5.1.2/La fuzzification [3]

La fuzzification est le premier traitement qui entre en compte dans la structure d'un contrôle régi par logique floue, durant cette phase les informations issues du système sont tout d'abord normalisées. Ensuite, les données normalisées sont transformées en qualifications linguistiques.

Les entrées dans un système flou sont en général mesurées à l'aide d'organes de mesures qui sont le plus souvent de type analogique. Étant donné que, l'implémentation du système flou se fait toujours en numérique, il faut d'abord convertir les entrées analogiques en digital, puis on procède à la fuzzification. L'adaptation des entrées permet de garantir l'appartenance de celles-ci à l'univers de discours choisi. La dernière étape de fuzzification est la génération des degrés d'appartenance à chaque valeur linguistique définie par sa fonction d'appartenance.

### II.5.5.1.3/Mécanisme d'inférence

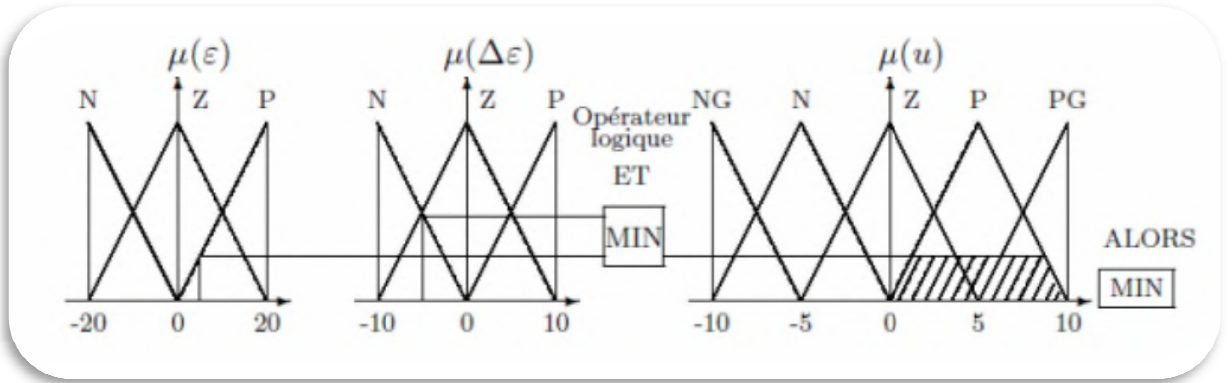
Considéré comme le « cerveau » du contrôleur, il permet de lier les degrés d'appartenance des fonctions d'appartenance d'entrée aux fonctions d'appartenance de sortie. Le degré d'appartenance de la fonction de sortie peut être calculé par différentes méthodes :

- La méthode d'inférence min-max (Mamdani).
- La méthode d'inférence max-prod (Larsen).
- La méthode d'inférence somme-prod (Sugeno).

#### ➤ La méthode Min-Max

C'est la méthode la plus universelle mais qui n'est guère applicable en raison du temps de calcul très long. Cette méthode réalise, au niveau de la condition, l'opérateur OU par la formation maximum et l'opérateur ET par la formation du minimum. La conclusion dans chaque règle, introduite par ALORS, lie le X facteur d'appartenance de la condition avec la fonction d'appartenance de la variable de sortie par l'opérateur ET, réalisé dans le cas présent par la formation du minimum.

Enfin l'opérateur OU qui lie les différentes règles est réalisé par la formation du maximum.



**Figure II.21 :** Définition de Alors dans la méthode d'inférence max-min

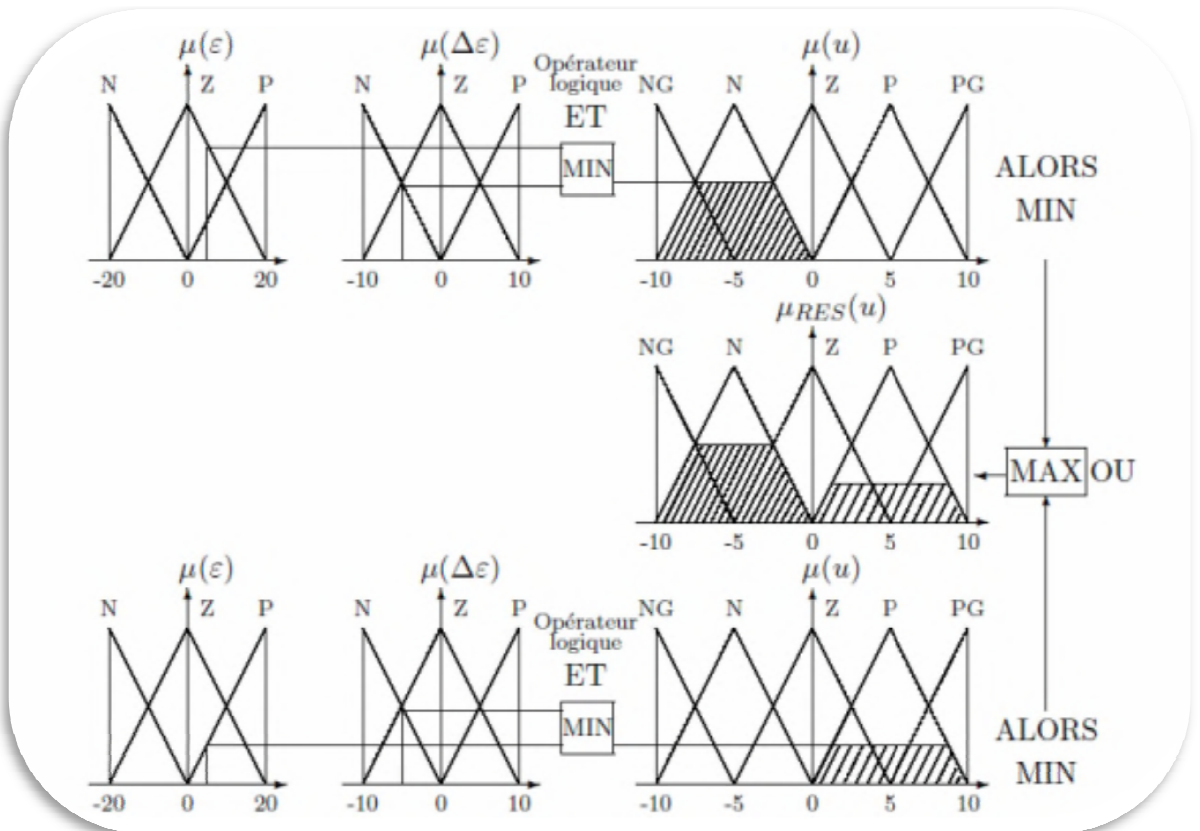
Par ailleurs l'interaction entre les règles sera nommée OU ainsi, on considère que les deux règles suivantes :

- Si  $\epsilon$  est et  $\Delta\epsilon$  est N ALORS u est N
- Si  $\epsilon$  est P et  $\Delta\epsilon$  est N ALORS u est P

Se lisent

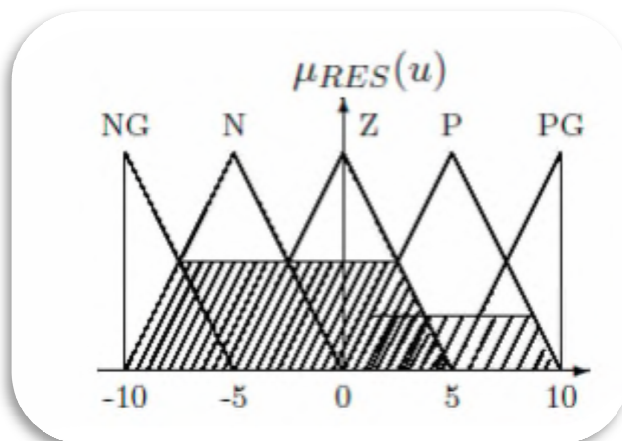
- Si  $\epsilon$  est Z et  $\Delta\epsilon$  est N ALORS u est N OU
- Si  $\epsilon$  est P et  $\Delta\epsilon$  est N ALORS u est P

Et ce OU se traduit par l'opération max, Nous obtenons ainsi la fonction d'appartenance résultant.



**Figure II.22 :** Définition de OU dans la méthode d'inférence max-min

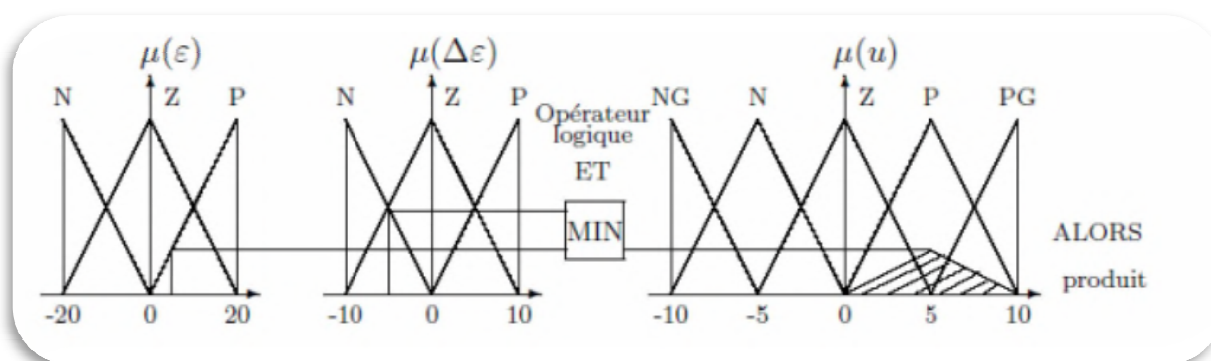
L'application de l'ensemble des règles donne la fonction d'appartenance partielle représentée sur la **Figure (I-23)**.



**Figure II.23 :** Fonction d'appartenance résultante

➤ **La méthode d'inférence max-prod**

La différence avec la méthode précédente est la réalisation de ALORS qui se traduit par la multiplication de la fonction d'appartenance considérée par la valeur de la règle. Ce résultat est illustré sur la **figure (II-24)**.



**Figure II.24:** Définition de ALORS dans la méthode d'inférence max-prod.

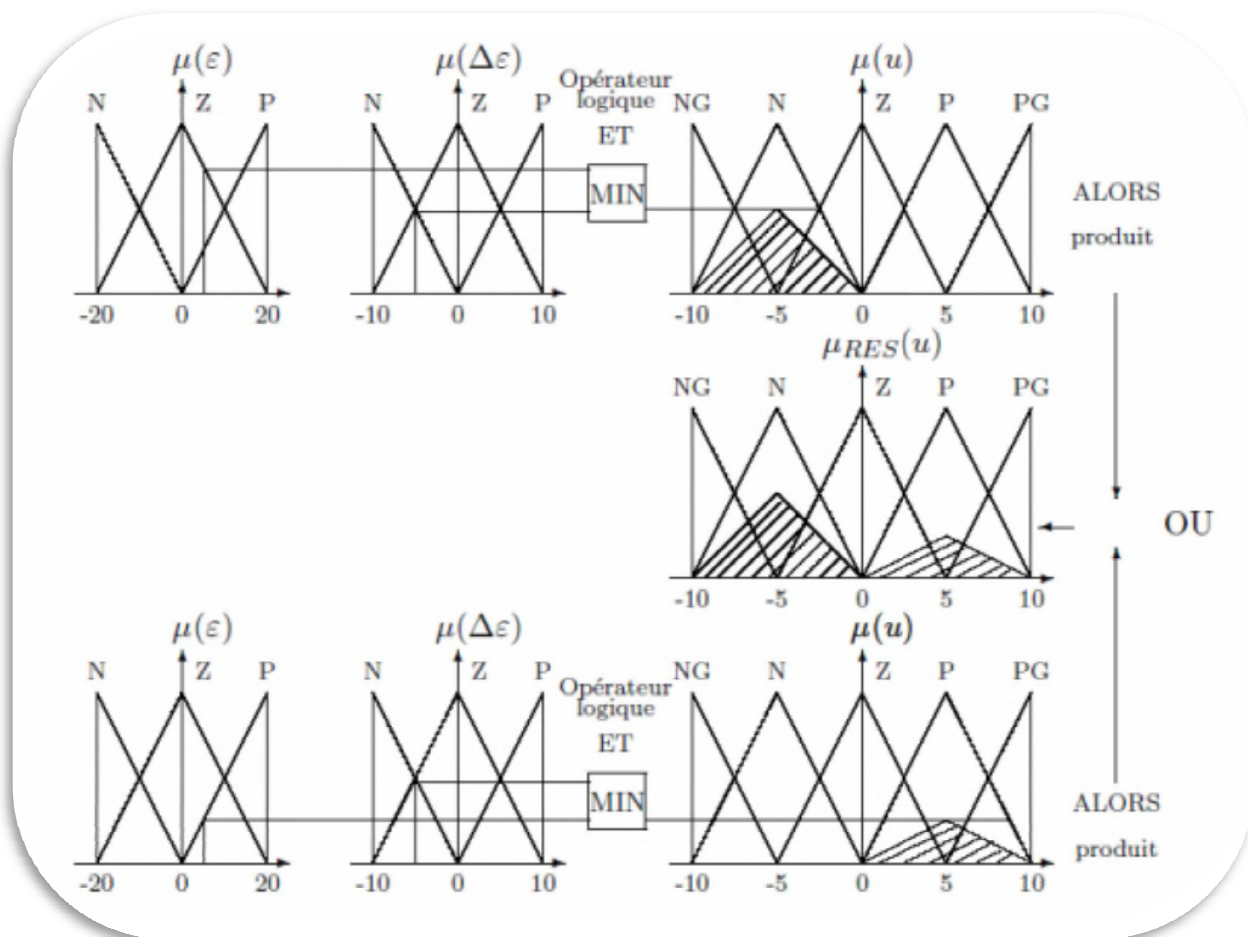


Figure II.25 : Définition de OU dans la méthode d'inférence

➤ **La Méthode d'inférence somme-prod**

Proche de la méthode précédente, il suffit de remplacer la traduction de ALORS par la somme des fonctions d'appartenance partielles.

Consiste à transformer l'ensemble flou de sortie résultant de l'agrégation des règles en une grandeur de commande déterministe et précise à appliquer au processus. Dans la littérature, il existe plusieurs stratégies pour réaliser cette opération telles que :

- Défuzzification par calcul du centre de gravité (barycentre).
- Défuzzification par calcul du maximum.

La méthode de Défuzzification par le centre de gravité, est la méthode la plus utilisée en commande floue du fait qu'elle fournit intuitivement la valeur la plus représentative de l'ensemble flou issu de l'agrégation des règles.

### ❖ Méthode par centre de gravité

Cette méthode est la plus coûteuse en temps de calcul mais la plus utilisée, elle consiste à prendre comme valeur de sortie l'abscisse du centre de gravité de la surface de la fonction d'appartenance résultante.

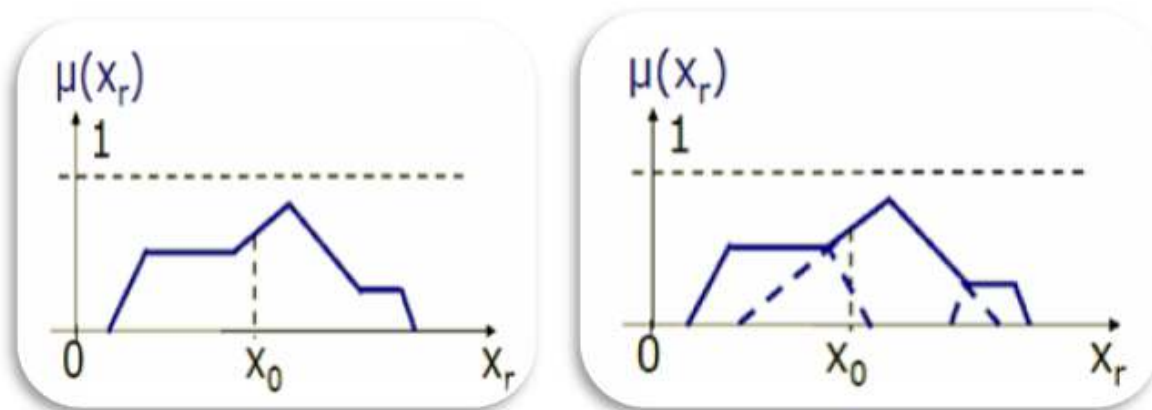


Figure II.26 Méthode par centre de gravité

### ❖ Méthode du maximum

Cette méthode consiste à choisir comme valeur de sortie correspondant à l'abscisse le maximum de la fonction d'appartenance.

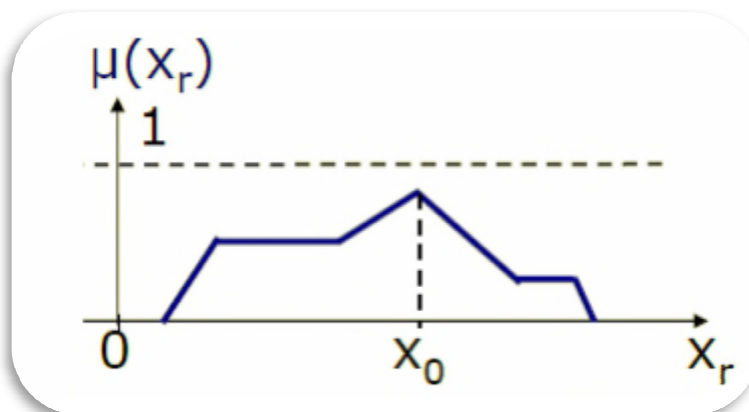


Figure II.27 : Méthode du maximum.

## II.5.6/ Etapes de 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.

### II.5.6.1/ Choix des variables d'entrées et de sorties

Pour le réglage de la vitesse de la machine courante à continu, nous avons choisi comme variables d'entrées l'erreur 'e' qui est la différence entre la vitesse de référence  $\Omega_{ref}$  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. Comme variable de sortie, une seule variable de commande qui est le couple de référence ( $c_{ref}$ ) est choisie.

### II.5.6.2/ Normalisation des variables d'entrée et de sortie

La normalisation se fait par une division des variables e,  $\Delta e$  et U 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.

### II.5.6.3/ 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 [-1 1], pour la dérivée de l'erreur entre [-1 1] et pour la sortie entre [-1 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 et trapézoïdale avec intersection de 50 % placées symétriquement par rapport à zéro sont utilisées. Le nombre des ensembles flous est cinq pour toutes les variables linguistiques qui sont successivement notées.

**NG** : Négatif Grand ; **N** : Négatif ; **Z** : Zéro ; **P** : Positif ; **PG** : Positif Grand.

### II.5.6.4/ 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  $\Delta e$  est  $x_2$  ALORS réf C est  $x_3$

Le nombre des ensembles flous pour e et  $\Delta e$  est cinq, donc il faut  $5 \times 5 = 25$  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 :

e	NG	N	Z	P	PG
$\Delta e$					
NG	NG	NG	N	N	Z
N	NG	N	N	Z	P
Z	N	N	Z	P	P
P	N	Z	P	P	PG
PG	Z	P	P	PG	PG

**Tableaux II.3 :** Règles à cinq fonctions d'appartenance

#### II.5.6.5/ 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.

#### II.5.6.6/ 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.

### II.5.7/ Application de la commande floue sur le moteur à courant continu

Après avoir énoncé et défini les concepts de base, les termes linguistiques utilisés en logique floue et la structure d'un régulateur flou. On s'intéresse donc à implémenter la commande floue pour un moteur à courant continu, pour un seul objectif est d'annuler l'erreur statique, diminuer le dépassement, diminuer le temps de réponse et le temps de montée afin d'obtenir une réponse adéquate du procédé et de la régulation et d'avoir un système précis, rapide, stable et robuste.

### II.5.7.1/Conception de contrôleur

Le toolbox fuzzy logic de MATLAB possède un éditeur qui permet de créer des systèmes d'inférences floues : des FIS (pour fuzzy inference system). Cette boîte à outils permet de générer des fichiers « .fis », qui correspondent à des systèmes d'inférences floues et dont font partie les RLF. Cette boîte à outils possède 3 éditeurs (de fis, de règles et de fonctions d'appartenance) qui permettent de saisir l'ensemble des données du FIS ainsi que deux interfaces graphiques qui permettent de visualiser les inférences directement sur la base de règles, ainsi que des surfaces de contrôle. Les fonctions d'appartenances sont du type triangulaire. L'intervalle d'intérêt des variables d'entrées est subdivisé en cinq classes pour l'erreur «e» et en cinq classes pour sa dérivée «de/dt», alors que celui de la variable de sortie «u» est subdivisé en cinq classes

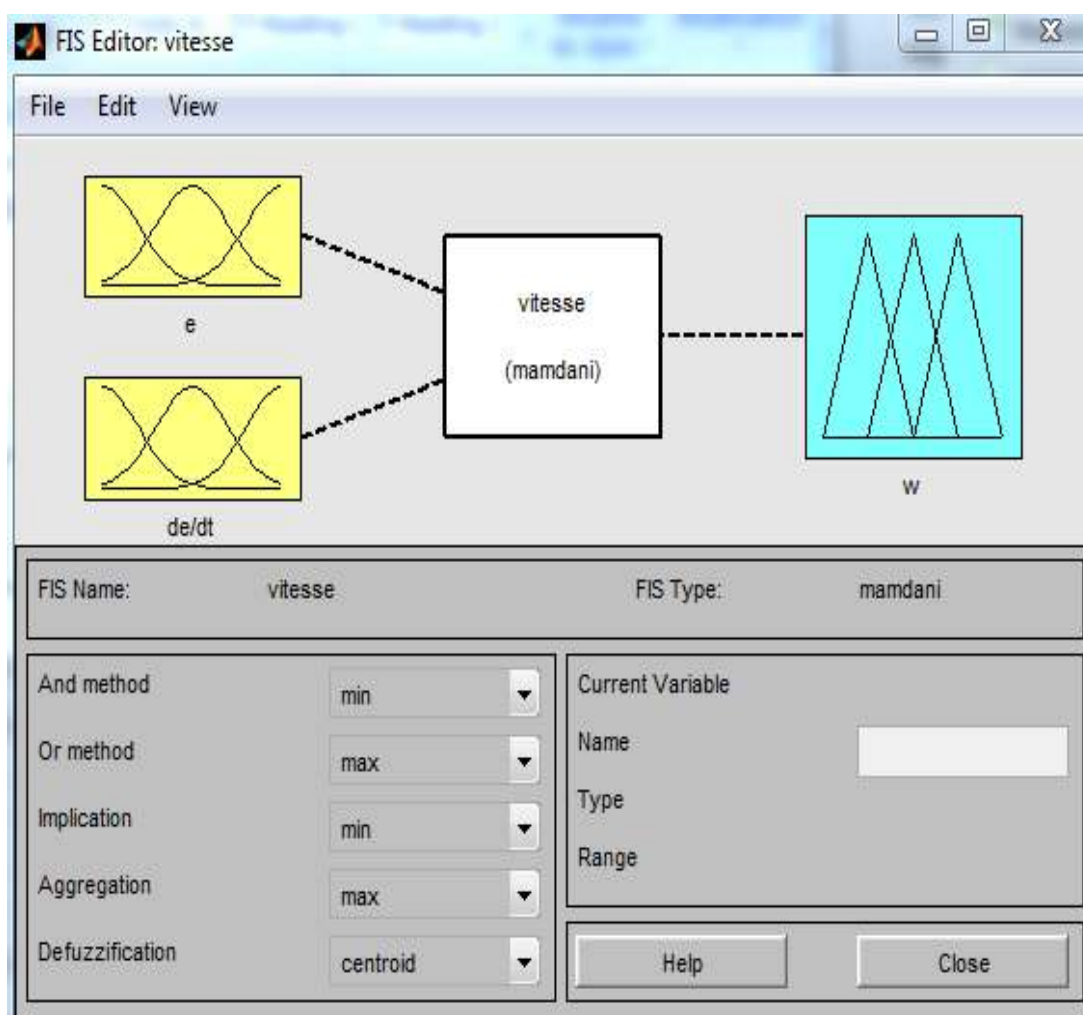


Figure II.28: Définition de la structure globale

Grâce à cette interface, on définit les plages d'entrées et de sorties de façon à contraindre le comportement du bloc Fuzzy, illustré par la Figure II.29, pour définir ces fonctions :

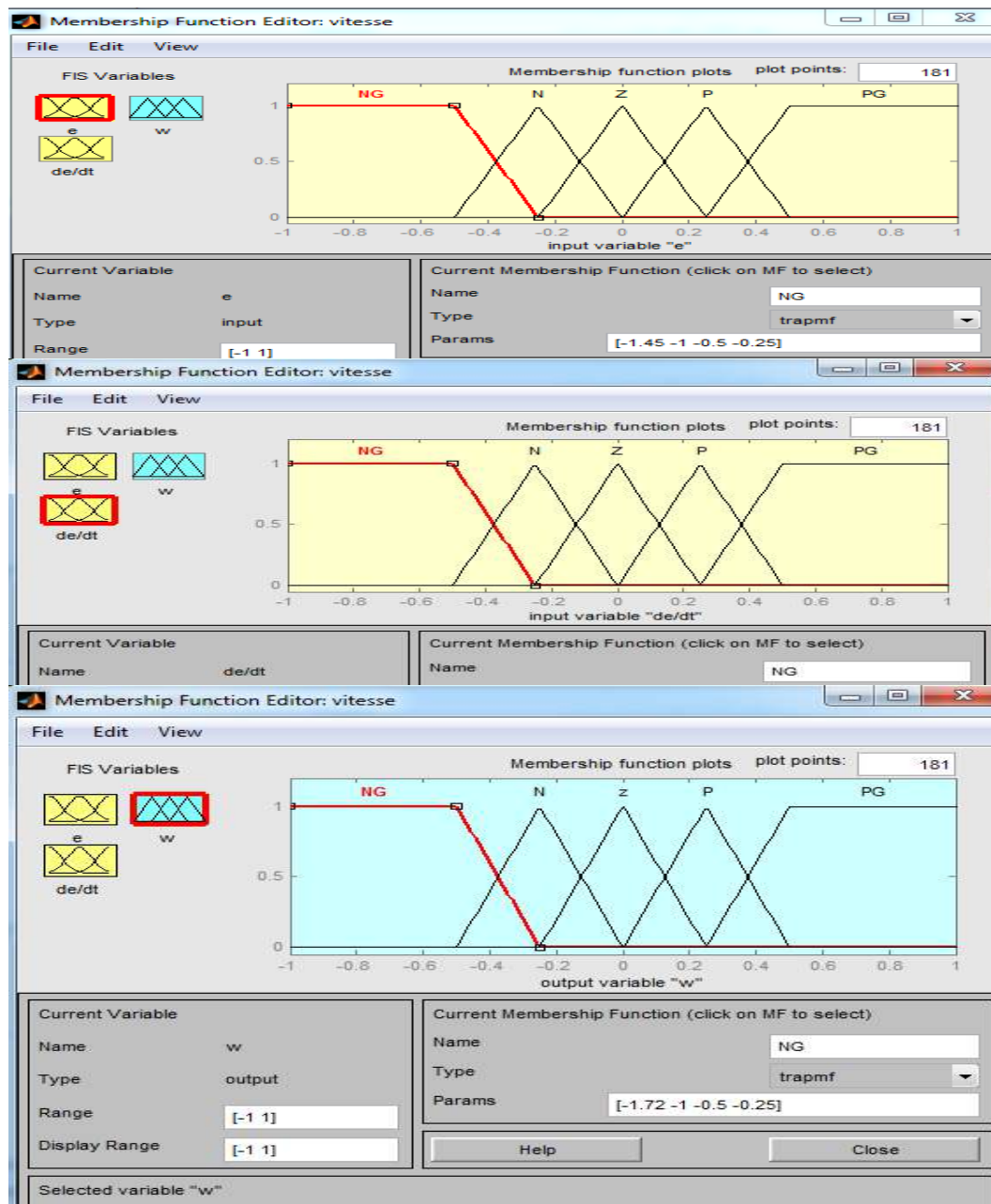
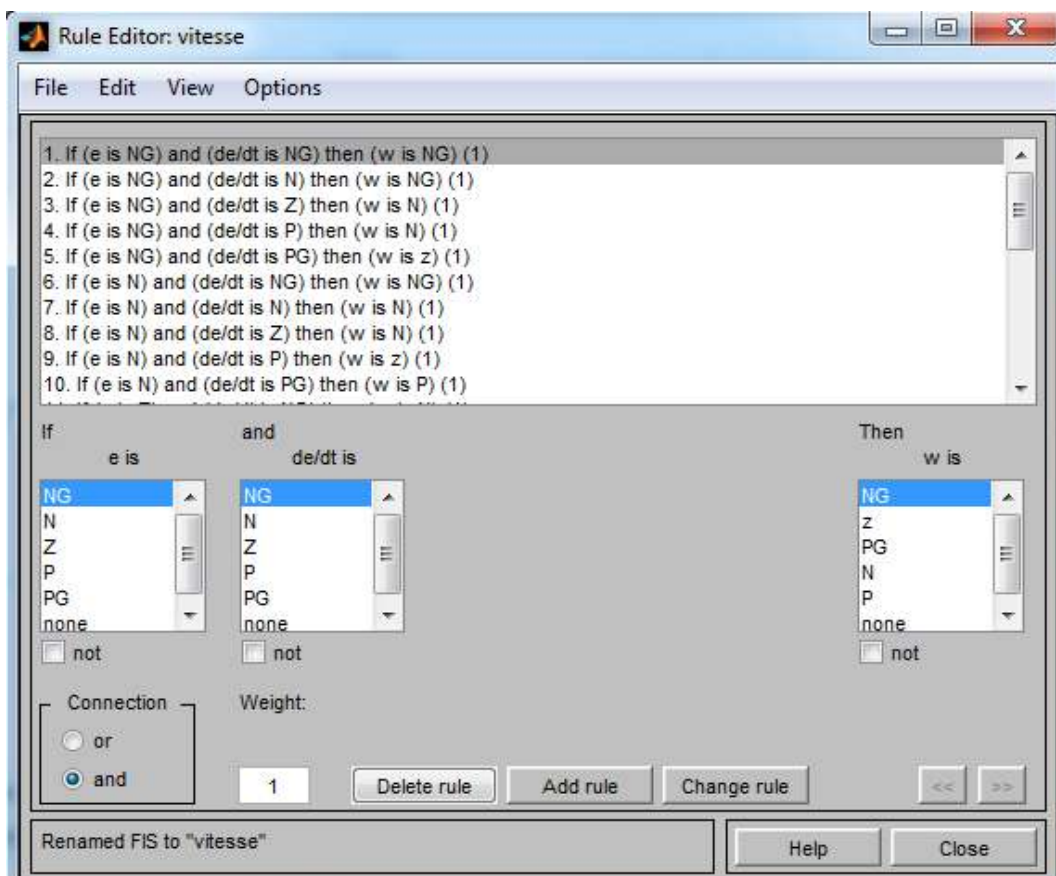


Figure II.29: Définition des fonctions d'appartenance pour 'e', 'de/dt' et 'u'

Il est également possible de configurer les lois du bloc flou, le principe celui d'actions conditionnelles «if » représenté sur la Figure II.30.



**Figure II.30:** Écran de saisie des règles

La sortie est ensuite déterminée suivant les poids respectifs des différentes lois d'entrées. En effet les lois se chevauchent, et donc une valeur de sortie est fonction de plusieurs lois d'entrées sur une même grandeur.

### II.5.7.2/ Schéma bloc de simulation de la MCC avec le contrôleur floue :

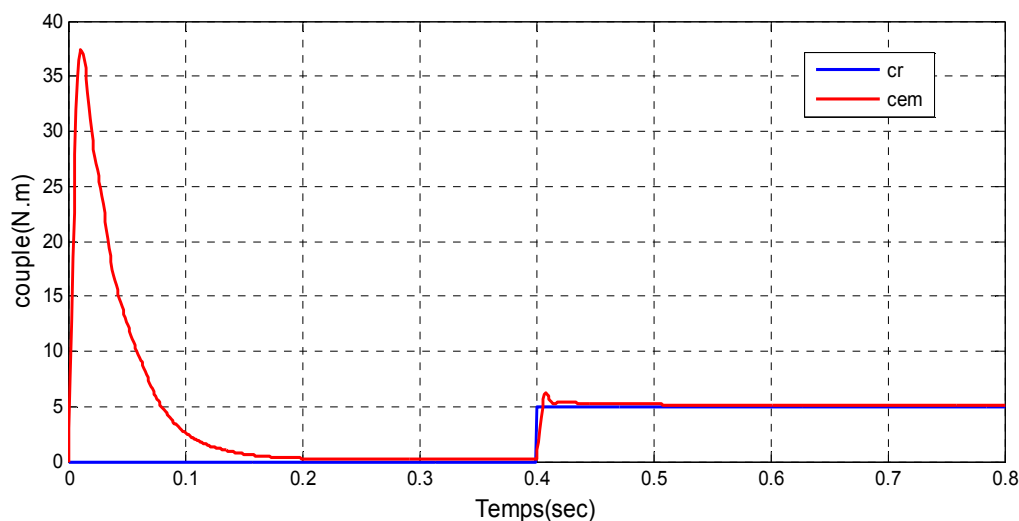
❖ **Régulation en vitesse :**

Pour illustrer les performances de réglage par la logique floue, nous avons simulé le schéma bloc de la commande en vitesse d'un moteur à courant continu qui est représenté par la Figure II.31, on conserve le régulateur de courant classique

Les valeurs des gains du régulateur sont choisis après plusieurs tests d'ajustement

$$Ke = \frac{1}{250}; K\Delta e = 1/9 \times 10^3; u = 25$$





**Figure II.33 :** Réponses de couple résistant( $C_r$ ) et couple électromagnétique( $C_{em}$ ) par logique floue

#### ❖ **Interprétation :**

L'allure de la vitesse possède une caractéristique presque linéaire et atteint la vitesse de référence dans un temps de réponse très petit temps de réponse  $t_r = 0.15$  sec et un temps monte  $t_m = 0.8$  sec avec un dépassement de 0% après l'application de la charge à l'instant  $t=0.3$  sec de 5Nm, on ne constate presque aucune influence sur l'allure de la vitesse, le courant subit au moment du démarrage un pic puis atteint rapidement la valeur nulle, au moment d'application la charge le courant à augmenter pour maintenir la charge (couple résistant)

#### ➤ **Remarque :**

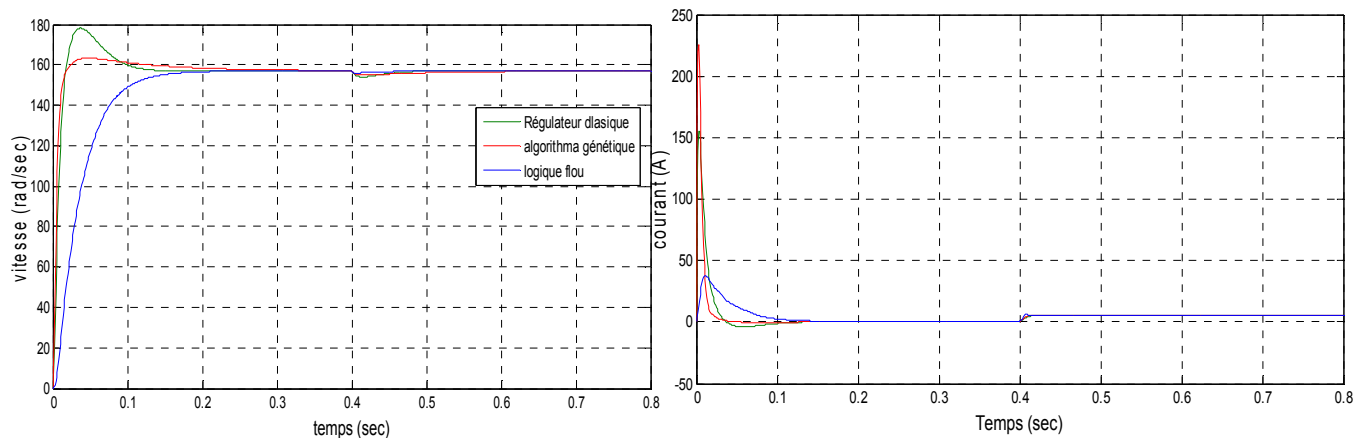
On remarque qu'à chaque fois on intervient sur les gains d'entrées de notre RLF on agit sur la rapidité du système et à chaque fois qu'on intervient sur le gain de la sortie du système on agit sur la précision

#### **II.5.8/ Conclusion :**

Les résultats de la simulation nous ont permis de juger les qualités de la commande à base du régulateur flou. A travers les caractéristiques de la réponse, on observe de bonnes performances même en présence des perturbations grâce à la robustesse du régulateur flou, Ce qui nous a permis de conclure que la logique floue est un moyen très performant pour l'amélioration de la stabilité du moteur à courant continu.

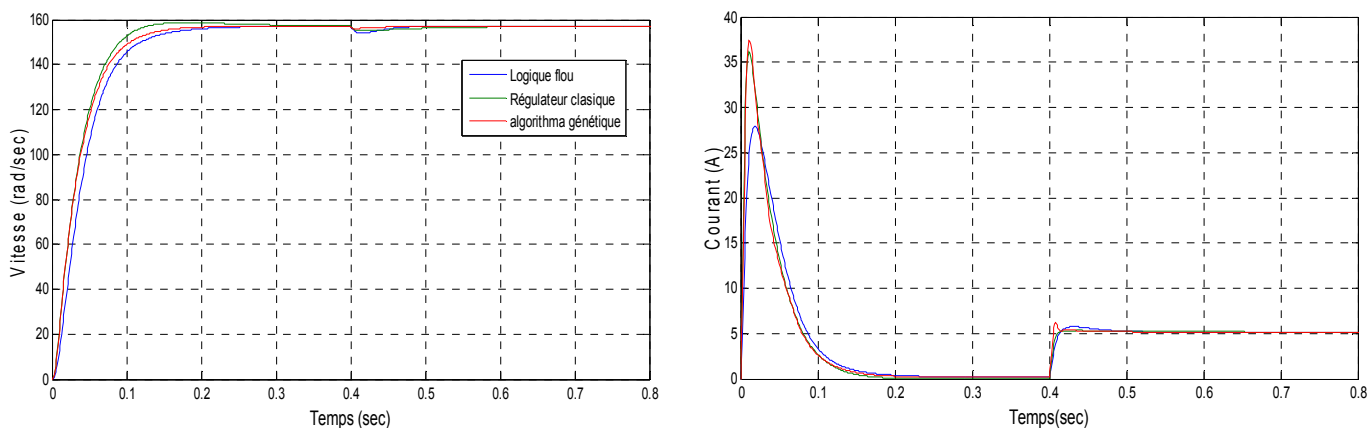
## II.6/ Comparaison entre les différentes méthodes de régulation

Simulation Sans filtre :



**Figure II.34** : Allure finale de la vitesse et du courant (Comparaison entre les différentes méthodes PI, BLF, AG)

Avec filtre



**Figure II. 35** : Allures de la vitesse et du courant avec filtre

### ➤ Interprétation

Si on compare les résultats obtenus avec le régulateur à algorithmes génétiques aux ceux de la logique floue et régulateur classique PI

Les résultats de simulation des algorithmes génétique et la logique floue obtenus montrent l'amélioration de la réponse dynamique de la vitesse. Cette dernière atteint sa référence en un temps de réponse (temps de réponse AG 0.015Sec et logique floue 0.09Sec).

De plus, l'allure de courants marque de pic respectivement inférieurs à ceux marqués dans le cas du PI classique. Application d'un couple de charge nominal Lorsque nous adoptons les méthodes de commande en vitesse intelligent, nous obtenons une réponse de vitesse qui suit parfaitement la référence demandée. Nous voyons que les impacts du couple de charge n'affectent pas la réponse en vitesse du système.

## II.7/ Conclusion

Ce chapitre constitue une contribution à la mise en œuvre des techniques non conventionnelles de commande numérique. Nous avons ainsi élaboré deux stratégies de contrôle de vitesse de machines à courant continu basées sur l'algorithme génétique et la logique floue . 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 à courant continue utilisant l'algorithme génétique ou la logique floue présentent des performances intéressantes

Diverses méthodes permettent d'optimiser les coefficients du PI. Mais le régulateur idéal n'existe pas, chaque optimisation d'un paramètre (précision, ...) se fait aux dépens d'un autre. Il convient alors de trouver le meilleur compromis en fonction des exigences initiales du cahier de charge.

# Chapitre III

Commande d'une MCC par l'optimisation par  
essaim particulaire (PSO)



### III.1/ Introduction

L'optimisation par essaim particulaire (OEP) ou bien (PSO), comme les algorithmes génétiques, est une méthode d'optimisation heuristique basée sur la simulation du comportement collectif des êtres vivants tels que des oiseaux ou des abeilles. Cette méthode d'optimisation, inventé par l'électricien Russel Eberhart et le socio psychologue James Kennedy en 1995, s'appuie notamment sur le modèle développé par le biologiste Reynold C.W. permettant de simuler le déplacement d'un groupe d'oiseaux. Cette méthode se base sur la collaboration des individus d'un même essaim en essayant de maintenir constante la distance qui les sépare afin d'éviter de se chevaucher lorsqu'ils changent de direction.

Cette technique est souvent décrite comme une sorte d'algorithme évolutionnaire avec une population d'individus (les particules) dans laquelle, à chaque pas de temps, les meilleurs sont plus au moins imités par les autres. Un autre aspect essentielle, propre à cette technique, est l'existence d'une mémoire que doit posséder chaque élément de l'essaim lui permettant de se souvenir de sa meilleure performance et celle transmise par ses congénères. De plus, les individus de l'essaim travaillent en collaboration (en s'échangeant des informations) et non pas en compétition comme dans les algorithmes génétiques par exemple.

Dans ce chapitre, on présente dans la première partie l'algorithme d'Optimisation par Essaim Particulaire et dans la seconde, on propose une application à cet algorithme pour le dimensionnement du régulateur proportionnel intégral utilisée pour la commande en vitesse de la machine à courant continu.

### III.2.Optimisation Par Essaims de Particule « PSO »:

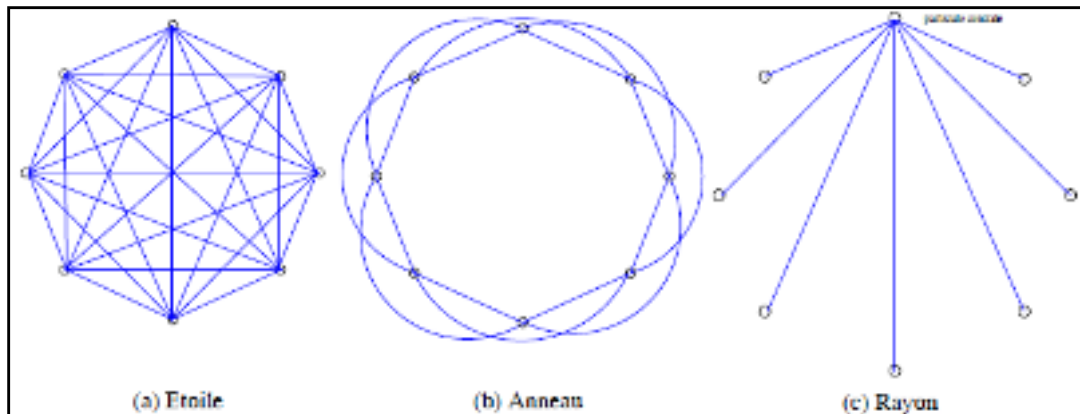
L'optimisation par Essaim de particule (OEP) ou bien (PSO Particle swarm optimization) est une technique utilisée pour explorer l'espace de recherche d'un problème quelconque pour trouver l'ensemble des paramètres qui maximise/minimise un objectif particulier. Cet objectif est atteint en suivant un algorithme dédié que l'on verra par la suite.

### III.3.Notion de voisinage :

Le voisinage constitue la structure du réseau social. Les particules à l'intérieur d'un voisinage communiquent entre-elles. En général, pour une nuée d'oiseaux, le voisinage suit trois types de topologies :

- Topologie en étoile (Figure1 (a)) : le réseau social est complet, donc une communication complète et une attirance vers la meilleure particule.

- Topologie en anneau (Figure1 (b)) : chaque particule communique avec  $n$  voisines immédiates. Chaque particule tend à se déplacer vers la meilleure dans son voisinage local.
- Topologie en rayon (Figure1 (c)) : une particule "centrale" est connectée à toutes les autres. Seule cette particule centrale ajuste sa position vers la meilleure, si cela provoque une amélioration l'information est propagée aux autres.



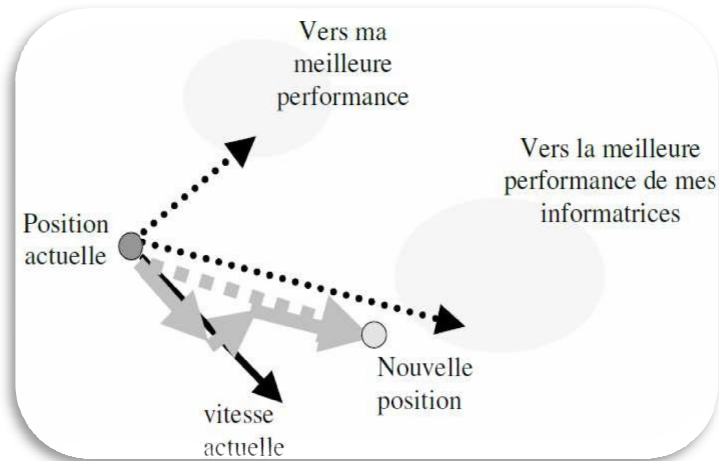
**Figure III.1 :** différents types de topologie pour un essaim de particules.

1 : L'optimisation est le mécanisme par lequel on trouve la valeur maximale ou minimale d'une fonction. Ce mécanisme est utilisé dans plusieurs domaines tels que: la physique, chimie, l'économie, et l'informatique...etc. où le but est de maximiser l'efficacité, la productivité et d'autre mesures.

#### III.4 L'ALGORITHME PARTICLE SWARM OPTIMIZATION

Chaque particule représente une solution potentielle dans l'espace de recherche. La nouvelle position d'une particule est déterminée en fonction de sa propre valeur et celle de ses voisines.

Le principe de la méthode d'essaim de particule est résumé par la figure (III.1). Pour réaliser son prochain mouvement, chaque particule combine trois tendance: suivre sa vitesse propre, revenir vers sa meilleure performance, aller vers la meilleure performance de ses informatrices



**Figure III.2.** Schéma de principe du déplacement d'une particule.

Un coefficient de confiance est alors associé à chacune de ces trois vitesses. Ainsi, la particule ne rejoint aucune des trois positions précédentes mais se déplace vers une nouvelle position qui est la combinaison linéaire de ces trois positions.

Soit  $\vec{x}_i(t)$  la position de la particule  $P_i$  au temps  $t$ , sa position est modifiée en ajoutant une vitesse  $\vec{v}_i(t)$  à sa position courante :

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (\text{III.1})$$

La vitesse de chaque particule est mise à jour suivant l'équation suivante:

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 [x_{pi}(t) - x_i(t)] + c_2 r_2 [g(t) - x_i(t)] \quad (\text{III.2})$$

$v_i(t)$  est la vitesse de particule  $i$  à l'instant  $t$  et  $x_i(t)$  est la position de particule  $i$  à l'instant  $t$ , les paramètres  $w$ ,  $c_1$ , et  $c_2$  ( $0 \leq w \leq 1.2$ ,  $0 \leq c_1 \leq 2$ , et  $0 \leq c_2 \leq 2$ ) sont des coefficients constants fixés par l'utilisateur,  $r_1$  et  $r_2$  sont des nombres aléatoires tirés à chaque itération,  $g(t)$  est la meilleure solution trouvée jusqu'à l'instant  $t$  et  $x_{pi}(t)$  est la meilleure solution trouvée par le particule  $i$ . C'est le vecteur vitesse qui dirige le processus de recherche et reflète la "sociabilité" des particules.

#### **III.4.1 Les variables et paramètres de l'algorithme :**

N nombre de particules

$\vec{x}_i(t)$  Position de la particule  $P_i$

$\vec{v}_i$  vitesse de la particule  $P_i$

$pbest_i$  Meilleure fitness obtenue pour la particule  $P_i$

$\vec{x}_{pbest_i}$  Position de la particule  $P_i$  pour la meilleure fitness

$\vec{x}_{gbest_i}$  Position de la particule ayant la meilleure fitness de toutes

$\rho_1, \rho_2$  valeurs aléatoires positives

Initialisations :

Initialiser aléatoirement la population.

Traitement :

**Répéter**

**Pour i de 1 à N faire**

**Si** ( $F(\vec{x}_i) > pbest_i$ ) **Alors**

$pbest_i \leftarrow F(\vec{x}_i)$

$\vec{x}_{pbest_i} \leftarrow \vec{x}_i$

**Fin Si**

**Si** ( $F(\vec{x}_i) > gbest_i$ ) **Alors**

$gbest_i \leftarrow F(\vec{x}_i)$

$\vec{x}_{gbest_i} \leftarrow \vec{x}_i$

**Fin Si**

**Fin Pour**

**Pour i de 1 à N faire**

$\vec{v}_i \leftarrow \vec{v}_i + \rho_1(\vec{x}_{pbest_i} - \vec{x}_i) + \rho_2(\vec{x}_{gbest_i} - \vec{x}_i)$

$\vec{x}_i \leftarrow \vec{x}_i + \vec{v}_i$

**Fin Pour**

**Jusqu'à ce que** (le processus converge)

La variation de la vitesse est proportionnelle à l'éloignement d'une solution par rapport à la solution globale.

Les variables aléatoires  $\rho_1$  et  $\rho_2$  peuvent être définies de la façon suivante :

$$\begin{cases} \rho_1 = r_1 c_1 \\ \rho_2 = r_2 c_2 \end{cases}$$

$r_1$  et  $r_2$  suivent une loi uniforme sur  $[0..1]$  et  $c_1$  et  $c_2$  sont constantes et représentent une accélération positive, avec  $c_1 + c_2 < 4$ .

Le critère de convergence peut être un nombre fixe d'itérations, suivant la fitness ou bien la variation lorsqu'elle tend vers 0.

On remarque qu'il y a Six paramètres qui rentrent en ligne de compte :

1. La dimension du problème.
2. Le nombre de particules.
3. Les valeurs des coefficients  $\rho$ .
4. La taille du voisinage.

5. La vitesse maximale.
6. L'inertie.

La vitesse peut être limitée par une vitesse maximale  $V_{max}$  et une vitesse minimale  $V_{min}$  pour éviter que les particules se déplacent trop rapidement ou trop lentement d'une région à une autre dans l'espace de recherche.

Un facteur d'inertie  $\Phi$  peut être appliqué à la vitesse Pour contrôler l'influence de celle-ci.

#### III.4.2/ Initialisation de l'essaim

L'initialisation de l'essaim consiste à fixer aléatoirement la position et la vitesse de chaque particule constituant l'essaim. Cette étape est commune à presque tous les algorithmes d'optimisation itératifs stochastique.

#### III.4.3/ Confinement d'intervalle

Initialement, chaque particule a sa propre vitesse et sa propre position limitée dans l'espace de recherche. A chaque itération, toutes les particules changent leur vitesse et se déplacent selon les équations (III.1) et (III.2) Certaines peuvent alors se déplacer hors de l'espace de recherche. Pour éviter ce problème, on assigne à la particule sortante la valeur du point de frontière le plus proche. La vitesse de la particule concernée est alors annulée pour l'empêcher, de se déplacer à la prochaine itération. En général, lorsque la solution trouvée par l'algorithme se trouve sur la limite de l'espace de recherche, cela signifie que quelques limites des paramètres ne sont pas correctes [5].

#### III.4.4/ Organigramme général d'un PSO

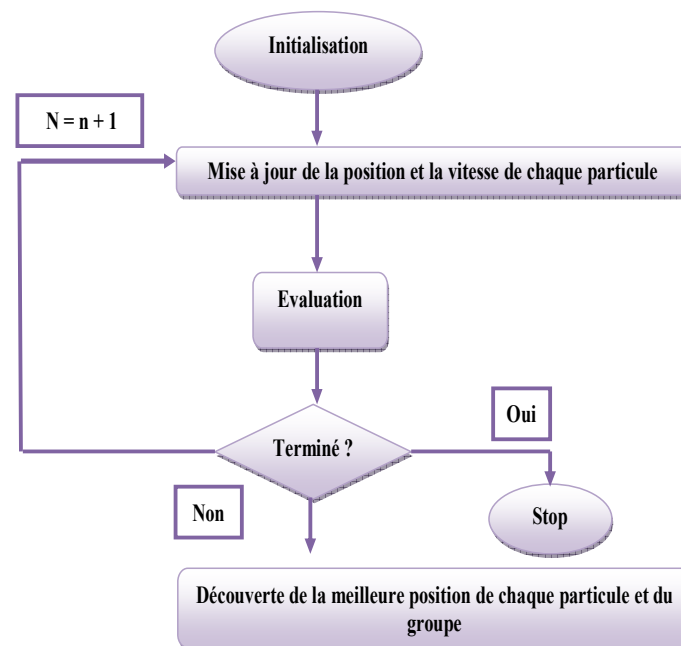


Figure III.3: Organigramme général de l'PSO

### III.5/ Avantages et inconvénients de PSO

L'algorithme du PSO présente un avantage certain sur les méthodes classiques dans le sens où il permet une exploration aléatoire de l'espace de solution. De plus de sa simplicité à mettre en œuvre et son exécution conduit à l'obtention de très bon résultat, rapidement et facilement avec peu de paramètre à ajuster.

Bien que le PSO trouve de bonnes solutions dans un temps beaucoup plus court que d'autres algorithmes évolutionnaires, l'amélioration de la qualité des solutions ne peut être garantie en augmentant le nombre d'itération.

L'algorithme de PSO a deux inconvénients principaux :

- Le premier inconvénient est que l'essaim peut prématurément converger.
- Le deuxième inconvénient est que les approches stochastiques ont un problème de dépendance tout changement d'un de leurs paramètres peut avoir un effet sur le fonctionnement de l'algorithme tout comme sur la solution obtenue.

### III .6 Optimisation des Gains du PI par la Technique PSO

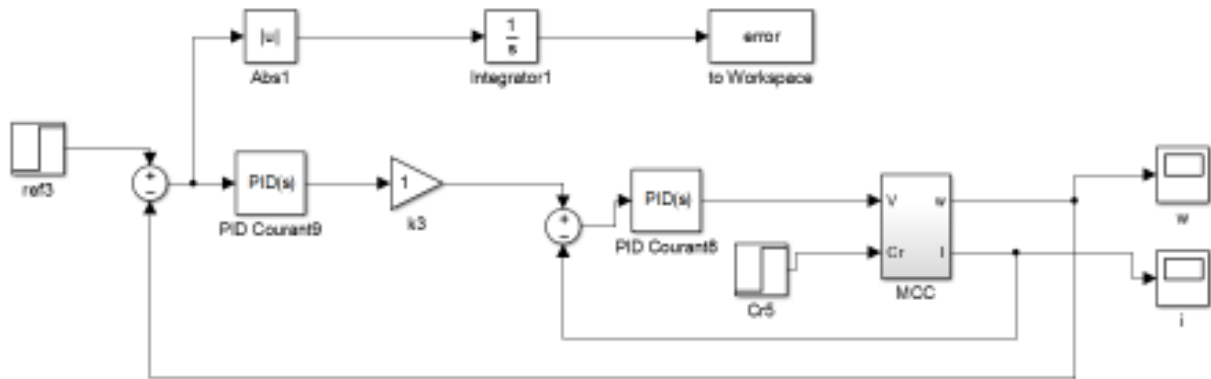
Le régulateur PI (Proportionnel, Intégral) est encore largement utilisé dans le milieu industriel malgré l'émergence d'autres méthodes de régulation. Ce régulateur linéaire est basé sur une structure très simple dont le fonctionnement ne dépend que de deux coefficients, qui sont les gains appliqués sur les signaux proportionnels ( $K_p$ ), intégral ( $K_i$ ). De nombreuses méthodes de réglage statique d'un PI ont été décrites dans la littérature, la plus connue étant certainement la méthode de Ziegler-Nichols [30], [31], [32]. Dans chacune de ces méthodes, les deux gains sont fixés en suivant une procédure de réglage qui garantit un fonctionnement optimal selon un ou plusieurs critères. Dans tous les cas, la fonction de transfert du régulateur PI reste linéaire.

Plus récemment, des auteurs ont proposé des méthodes de réglage dynamique des coefficients d'un PI et ils montrent que les performances sont d'autant meilleures qu'on utilise les méthodes classiques [30], telle que le correcteur flou « FLC » et des méthodes intelligentes tel que l'algorithme génétique (AG) et la technique PSO.

Pour notre étude nous avons choisi de minimiser l'erreur  $\varepsilon(t)$  entre la vitesse de référence et la vitesse réelle de la machine . La fonction à optimiser est donnée par l'équation :

$$\varepsilon(t) = \Omega_{\text{ref}} - \Omega \quad (\text{III.3})$$

Pour la mise en œuvre informatique nous avons exploité les équations (III.1) et (III.2) pour le développement d'un programme sous MATLAB qu'on peut le schématiser sous l'organigramme présenté par la figure (IV.2), avec l'intégration des PSO dans le schéma simulation de la MCC la figure (III.4).



**Figure III.4** Schéma bloc de simulation de la MCC avec le PSO

L'implémentation des PSO a été effectuée en se référant à l'organigramme suscité.

Mais avant tout, on donne quelques explications aux symboles utilisés :

$P_{bi}$  : représente la  $i$ ème meilleure solution (position) de la particule  $i$ .

$P_g$  : représente la meilleure solution (position) globale du groupe.

$X_i$  : représente la solution (position) de la  $i$ ème particule.

$V_i$  : représente la vitesse (vitesse) de la  $i$ ème particule.

$f$  : la fonction fitness à optimiser.

$\varphi_1, \varphi_2$  : sont des pondérations choisies généralement égales à 2.

$r_1, r_2$  : sont deux variables aléatoires uniformément distribuées.

$w$  : représente l'inertie est fixée à 0.8.

Il est à noter que chaque particule contient un nombre de composantes, pour notre exemple chaque particule contient deux composantes (coefficients des régulateurs) à savoir :  $(K_{pi}, K_{ii})$

L'organigramme se récapitule en :

**Étape 1** : initialisation de toutes les positions locales  $X_i$ , meilleures positions locales  $P_{bi}$  et la meilleure position globale  $P_g$ .

**Étape 2** : évaluer la fonction fitness de chaque particule  $f(X_i)$ .

**Étape 3** : mise à jour de toutes les meilleures positions locales  $P_{bi}$ .

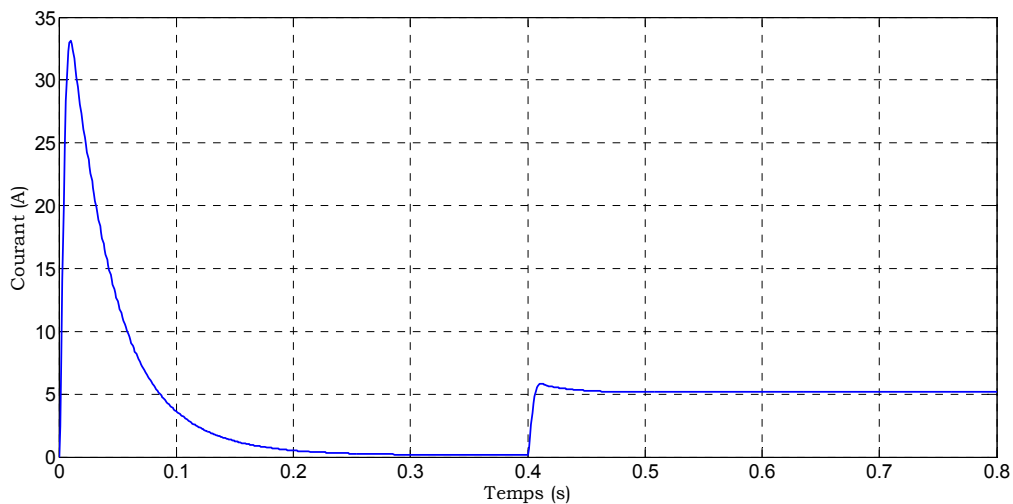
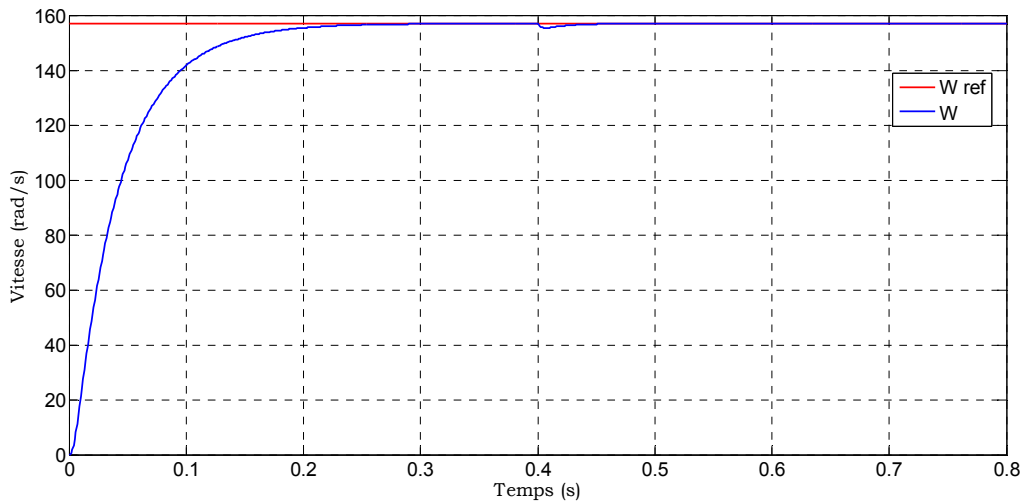
**Étape 4** : si  $f(P_{bi}) < f(P_g)$  alors  $P_g = P_{bi}$

**Étape 5** : mise à jour de  $X_i$  et  $V_i$  suivant les deux équations (III.1) et (III.2).

**Étape 6** : si le critère d'arrêt est satisfait alors stop si non aller à étape 2.

### III.7 Résultats de simulation :

La figure (VI.4) présente les réponses indicielles à une vitesse de référence  $\omega_{ref}=157\text{rad/sec}$  pour les valeurs  $k_p=-2.3877$  et  $k_i=70.9101$  trouvées par l'algorithme d'optimisation PSO. Figure III.5 nous permet de remarquer la performance de qualité de signal de vitesse, le courant, ainsi que le couple après la régulation par l'algorithme d'optimisation PSO (temps de réponse minimale, stabilisation) par rapport à la régulation de vitesse par PI (temps de montée lent, le signal prend un temps pour stabilisé), de même que les



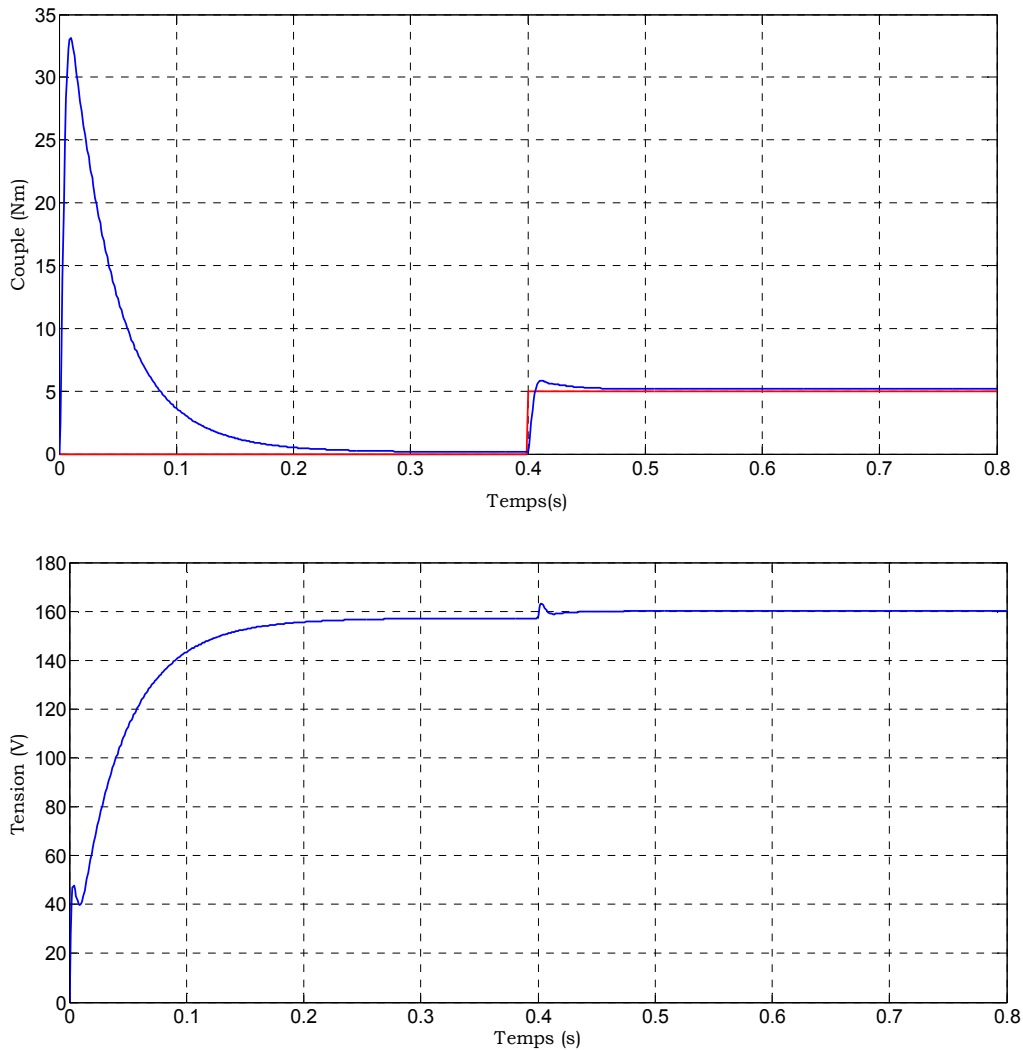


Figure III.5 réponse de régulation par l'algorithme d'optimisation PSO.

### III.8 CONCLUSION

Dans ce chapitre, une nouvelle approche a été proposée pour le dimensionnement du régulateur proportionnel intégral par la méthode d'optimisation par essaim particulaire. Cette approche a donné des résultats satisfaisants, et se distingue par sa simplicité, notamment par un programme composé d'un nombre assez réduit d'instructions. Contrairement à l'approche habituelle qui consiste à passer par plusieurs étapes et qui ne conduit pas directement à des paramètres optimaux (il a fallu changer les paramètres trouvés jusqu'à ce qu'on obtient le cahier des charges voulu).

# Conclusion générale

### Conclusion générale

Durant notre projet, nous avons vu la mise en œuvre des techniques non conventionnelles de commande numérique. Nous avons ainsi élaboré trois stratégies de contrôle de vitesse des machines à courant continu basées sur la logique floue, les algorithmes génétiques et l'optimisation par essaim particulaire. 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 la vitesse des MCC utilisant (RLF, AG, PSO) présentent des performances intéressantes.

La régulation de la machine prise comme un bloc à une grandeur d'entrée  $u(t)$  et une grandeur de sortie  $W(t)$  a permis d'obtenir les performances désirées mais elle a engendré un pic très important de courant au démarrage. La commande en cascade permet de contrôler non seulement la vitesse mais aussi le courant  $i(t)$  à d'où la possibilité de sa limitation.

La synthèse du régulateur par cette méthode présente des inconvénients. En effet, elle ne permet pas toujours l'obtention des paramètres satisfaisants du correcteur, dès le premier coup, cependant on retouchera à ces paramètres jusqu'à ce que le cahier des charges soit tenu. Cette méthode nécessite également la connaissance des formules pour le calcul des paramètres de régulateur. Enfin, on dira que telle méthode ne supprime pas l'intervention de l'homme pour la conception optimale des régulateurs.

## RESUME

L'application de la régulation de vitesse des moteurs électriques est très répandue dans les systèmes industriels et automatiques, l'un des critères d'efficacité d'une telle régulation est sa robustesse face aux changements. Elle permet d'effectuer une régulation des paramètres physiques (vitesse, position, température... etc.) des systèmes en boucle fermée, c'est à dire un système capable d'atteindre et de maintenir une certaine consigne en utilisant une mesure, dont il reste à déterminer la fonction permettant de corriger la commande en fonction de la consigne initiale et la valeur mesurée. Dans notre cas, le système étudié est un moteur à courant continu dont le but est d'annuler l'erreur statique, diminuer le dépassement, le temps de montée et le temps de réponse afin d'obtenir une réponse adéquate du procédé et de la régulation. L'objectif est d'obtenir un système précis, rapide, stable et robuste.

### الملخص:

يتم تلبية تطبيق تنظيم السرعة للمحركات الكهربائية بشكل جيد للغاية في الأنظمة الصناعية والأوتوماتيكية ، وأحد معايير الكفاءة لهذا التنظيم هو متانته في مواجهة التغيرات. يجعل من الممكن تنظيم المعلمات الفيزيائية (السرعة ، والموضع ، ودرجة الحرارة ، وما إلى ذلك) لأنظمة الحلقة المغلقة ، أي نظام قادر على الوصول إلى نقطة ضبط معينة والحفاظ عليها باستخدام قياس ، يكون الوظيفة التي تسمح بتصحيح عنصر التحكم كدالة لنقطة الضبط الأولية وتبقى القيمة المقاسة قيد التحديد. في حالتنا ، النظام المدروس هو محرك تيار مباشر ، والغرض منه إلغاء الخطأ الساكن ، لتقليل التجاوز ، ووقت الصعود ووقت الاستجابة من أجل الحصول على استجابة مناسبة من العملية و تنظيم. الهدف هو الحصول على نظام دقيق وسريع ومستقر وقوي.

## Références Bibliographiques

- [1]. M. Mohamed ARHOUIDAM, «mémoire mis en œuvre d'une régulation de vitesse du MCC», Université Mohammed Elkhamas, Maroc, 2013.
- [2]. P. Barret, 'Machines électriques', Edition Ellipses, 2002
- [3]. BENOUMESSAD Ismahane ET DJEBBAR Ahlem (ETUDE ET REALISATION D'UN HACHEUR SERIE POUR LA COMMANDE D'UN MOTEUR A COURANT CONTINU) mémoire de master option Electrotechnique industrielle, Université batna 2 2018.
- [4]. M. A. Johnson & Mohammed H.Moradi «PID Control (New identification and design methods) », Springer, 2005
- [5]. Freddy Mudry, Ajustage des paramètres d'un régulateur Pid, Département d'électricité et Informatique - Ecole d'ingénieur du Canton de Vaud, Mars 2006
- [6]. Réglages D'une Boucle Pid [En Ligne], The national institutes of standards and technology, Url =[Www.Linuxcnc.Org/Docs/Html/Motion/Pid\\_Theory\\_Fr.Html](http://Www.Linuxcnc.Org/Docs/Html/Motion/Pid_Theory_Fr.Html), Date de consultation le : 01/11/2015.
- [7]. ELMIR Younes, Optimisation multi objectif par les algorithmes génétiques et approche Pareto de paramètres d'un contrôleur PI- mémoire de Magistère - Université d'Oran Mohamed Boudia
- [8]. Lamia Mamlouk et Fouad Benouirane. "Vie Artificielle". 2003
- [9]. Laurent Bougrain. "Intelligence artificielle". Projet CORTEX, LORIA/INRIA LORRAINE, 2000
- [10]. D. Beasley, « An Overview of Genetic Algorithms Part1: Fundamentals », University of Cardiff, Cardiff, CF2, 4YN, United Kingdom, 1993.
- [11]. D.E. Goldberg, « Algorithmes génétiques. Exploration, optimisation et apprentissage automatique », Addison-Wesley. France, 1994.
- [12]. M. MITCHELL, « An Introduction to Genetic Algorithms », London: Cambridge, Massachusetts. London, England, 1999.
- [13]. David E.Goldberg. "Algorithmes génétiques Exploitation, optimisation et apprentissage automatique". Addison-Wesley, France, SA, 1994.
- [14]. Youssef Harrat. "Contribution à l'ordonnancement conjoint de la production et de la maintenance : Application au cas d'un job Shop". Thèse de Doctorat, L'U.F.R des Sciences et Techniques, Université de Franche-comté, 2003.

- [15]. Les critères d'arrêt se résument alors en : 1. Arrêt après un nombre de générations fixé à priori. 2. Arrêt lorsque la population cesse d'évoluer ou en présence d'une population homogène
- [16]. O. Landolt, P. Marchal et E. Vittoz, « processeurs flous », Eisi-Micro Swiss Semaine d'enseignement spécifique, Exposé on st-Inier, Décembre 2000.
- [17]. 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.
- [18]. R. Benabdennour, P. Borne, M. Ksouri, et F. M'sahli « Identification et Commande des procédés Industrielles », Edition technique, Paris 2001.
- [19]. LAOUFI Mourad, BABA Nawel, DJAMAI Hayet, Planification de la trajectoire D'un robot mobile par la logique floue. Mémoire de fin d'étude, l'université de Bechar Promotion 2010
- [20]. S.KHELKHAL, La logique floue et le contrôle avancé en milieu industriel, Mémoire de fin d'étude. Casablanca / Maroc .2004.
- [21]. GUENOUNOU Ouahib, Méthodologie De Conception De Controleurs Intelligents Par L'approche Génétique-application à un Bioprocédé, THESE De Doctorat De l'Université De Toulouse.2009.
- [22]. DJAABOUB Salim, Logique floue et SMA : Aide à la décision floue dans les systèmes multi- agents, Mémoire Magister en informatique. Universitaire Mentouri de Constantine 2009. [4] S.KHELKHAL, La logique floue et le contrôle avancé en milieu industriel, Mémoire de fin d'étude. Casablanca / Maroc .2004.
- [23]. Gonzalo Cabodevila, Commande et identification Floue. Ecole Nationale Supérieure de Microtechnique.1993.
- [24]. Olivier CAZIER, Instabilités dynamiques de systèmes frottants en présence de variabilités Paramétriques, Application au phénomène de crissement, Thèse de doctorat, l'université de VALENCIENNES.2012
- [26]. J. Kennedy et Eberhart R. " Particle Swarm Optimization". IEEE International Conference Neural Networks, vol, IV, pp.1942-1948 Perth, Australia, 1995.
- [27]. CW. Reynolds. " Flocks, herds and schools : a distributed behavioural model. Computer Graphics", vol. 21, n°4, pp. 2534, 1987.