

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE**



Université Mohammed Seddik BENYAHIA - Jijel

**Faculté des Sciences et de la Technologie
Département d'Electronique**

**Mémoire de fin d'études pour l'obtention du Diplôme de
Master en Electronique**

Option : Electronique des systèmes embarqués

Thème :

**Etude comparative en simulation entre un
régulateur PID et un régulateur flou**

Réalisé par :

- BOUTANA Walid
- YKHELFOUNE Nedjmeddine

Proposé par :

Dr. TALBI Nesrine

Promotion : Juillet 2019

Remerciements

Nous remercions Allah de nous avoir donné la volonté, la patience et le courage pour terminer ce travail.

Nous tenons à formuler notre gratitude à notre promoteur Madame « Dr. Talbi Nesrine » qui nous a honorés en dirigeant ce travail. Nous la remercions vivement pour sa disponibilité et ses conseils judicieux.

Nous remercions nos familles pour leur soutien moral durant notre projet de fin d'étude.

Nous remercions également le président de jury et les membres de jury qui ont accepté de juger ce travail.

Nous exprimons notre profonde gratitude au chef de département ainsi à tous les enseignants du département d'électronique.

Dédicace

Je dédie ce modeste travail en signe reconnaissance et de respect :

A mes chers parents

A ma cher sœur hanane

A mes frères Karim, salim et saadane

A ma cousine madame Hafssaoui Ilhem

A toute la famille qui mon donner de l'aide et de la sagesse

A mes amis Mohmoud, Abdeljalil, Housseem, Bachir.

A ma collègue Wazina

A tous mes collègues de la promotion Electronique des systèmes embarqués 2019.

A mon binôme Nedjm eddine.

Walid



Dédicace

Je dédie ce modeste travail en signe reconnaissance et de respect :

A mes chers parents

A mes sœurs

A mon frère

A toute la famille qui mon donner de l'aide et de la sagesse

A tous mes amis

A tous mes collègues de la promotion Electronique des systèmes embarqués 2019.

A mon binôme Walid.

Nedjm eddine



TABLE DES MATIERES

INTRODUCTION GENERALE	1
CHAPITRE I : LES REGULATEURS CONVENTIONNELS PID	
I.1 INTRODUCTION	3
I.2 NOTION DE SYSTEME	4
I.2.1 La boucle d’asservissement	4
I.2.1.1 Qualité d’un asservissement et performances d’un système	6
I.2.1.1.1 Stabilité.....	6
I.2.1.1.2 Rapidité	6
I.2.1.1.3 Précision	7
I.3 SYSTEMES ELEMENTAIRES	7
I.3.1 Notions de linéarité et non linéarité	7
I.3.2 Système du premier ordre	8
I.3.3 Système du 2 ^{ème} ordre	8
I.3.4 Système d’ordre supérieur à 2	9
I.4 LES REGULATEURS PID	10
I.4.1 Système de commande en boucle ouverte	10
I.4.2 Système de Commande en Boucle Fermée	10
I.4.3 Types de contrôleurs	11
I.4.3.1 Correcteur proportionnel (P)	12
I.4.3.2 Correcteur proportionnel – intégrale (PI)	12
I.4.3.3 Correcteur proportionnel – dérivée (PD)	13
I.4.3.4 Correcteur proportionnel - intégrale – dérivée	14
I.4.4 Les différentes structures du PID	15
I.4.4.1 La structure parallèle.....	15
I.4.4.2 La structure mixte.....	16
I.4.4.3 La structure série	16
I.5 METHODES DE SYNTHESE DU REGULATEUR PID	16
I.5.1 Méthode de Ziegler -Nichols.....	17
I.6 CONCLUSION	18

CHAPITRE II : LA COMMANDE FLOUE

II.1	INTRODUCTION	19
II.2	THEORIE DES SOUS-ENSEMBLES	20
II.2.1	Univers du discours	20
II.2.2	Variables linguistiques	20
II.2.3	Les règles linguistiques	21
II.2.4	Support	21
II.2.5	Hauteur	21
II.2.6	Noyau	22
II.2.7	Fonction d'appartenance	22
II.2.7.1	Fonction triangulaire	22
II.2.7.2	Fonction trapézoïdale	23
II.2.7.3	Fonction gaussienne	23
II.2.7.4	Fonction sigmoïde	24
II.2.8	Les opérations sur les ensembles flous.....	24
II.2.8.1	Egalité	24
II.2.8.2	Union.....	24
II.2.8.3	Intersection.....	24
II.2.8.4	Complément	25
II.2.8.5	Inclusion.....	25
II.3	LA COMMANDE FLOUE	25
II.3.1	Description générale d'un contrôleur flou.....	25
II.3.2	Interface de fuzzification.....	26
II.3.3	Base de connaissances	28
II.3.4	Moteur d'inférence floue.....	29
II.3.4.1	Méthode d'inférence max-min	30
II.3.4.2	Méthode d'inférence max-produit.....	30
II.3.4.3	Méthode d'inférence somme-produit	30
II.3.5	Interface de défuzzification.....	31
II.3.6	Les Différents types de contrôleurs flous.....	32
II.3.6.1	Le contrôleur flou e type Mamdani.....	32
II.3.6.2	Le contrôleur flou e type Takagi-Sugeno.....	33
II.4	AVANTAGES ET INCONVENIENTS DE LA LOGIQUE FLOUE	33
II.5	DOMAINES D'APPLICATIONS	34
II.6	CONCLUSION	34

CHAPITRE III : SIMULATION & INTERPRETATIONS

III.1	INTRODUCTION	35
III.2	APPLICATION DE LA COMMANDE PID	36
III.2.1	Système 1 ^{ier} ordre : Régulation de la température d'une serre	36
III.2.1.1	Présentation du système	36
III.2.1.2	Commande du système.....	36
III.2.2	Système de 2 ^{ème} ordre : Régulation d'un moteur à courant continu.....	38
III.2.2.1	Présentation du système	38
III.2.2.2	Commande du système.....	39
III.2.3	Système de 3 ^{ème} ordre :.....	41
III.2.3.1	Présentation et commande du système	41
III.3	APPLICATION DE LA COMMANDE FLOUE.....	43
III.3.1	Etapas pour créer le régulateur flou en utilisant la boîte à outil fuzzy-logic :.....	43
III.3.2	Système de 1 ^{ier} ordre : Régulation de la température d'une serre	46
III.3.2.1	Régulateur flou à une seule entrée	46
III.3.2.2	Régulateur flou à deux entrées	48
III.3.3	Commande Floue du système de 2 ^{ème} ordre	49
III.3.3.1	Commande du système en poursuite	50
III.3.3.2	Commande en régulation	51
III.3.4	Régulation du système 3 ^{ème} ordre à une seule entrée	52
III.3.4.1	Commande en poursuite du système	52
III.3.4.2	Régulation du système (avec perturbation)	53
III.4	COMPARAISON ENTRE LA COMMANDE FLOUE ET LA COMMANDE PID	54
III.5	CONCLUSION	58
	CONCLUSION GENERALE	59
	BIBLIOGRAPHIE.....	61
	ANNEXE.....	63

LISTE DES FIGURES

Fig. I.1 – Système de chauffage central	4
Fig. I.2 – Asservissement en boucle ouverte.....	5
Fig. I.3 – Asservissement en boucle fermée d’un système de chauffage.....	5
Fig. I.4 – Asservissement en boucle fermée.....	6
Fig. I.5 – commande en boucle fermée.	11
Fig. I.6 – Commande proportionnelle P.....	12
Fig. I.7 – Commande proportionnelle-intégrale PI.	13
Fig. I.8 – Commande proportionnelle-dérivée PD.....	14
Fig. I.9 – Commande PID	14
Fig. I.10 – La structure parallèle	15
Fig. I.11 – La structure mixte.....	16
Fig. I.12 – La structure série.	16
Fig. II.1 – Caractéristiques d’un sous-ensemble flou A	22
Fig. II.2 – fonction appartenance.....	23
Fig. II.3 – structure de base d’un contrôleur flou	26
Fig. II.4 – structure de base d’un contrôleur flou.....	27
Fig. II.5 – Fuzzification continue avec trois fonctions d'appartenance	27
Fig. III.1 –Méthodologie de commande.....	36
Fig. III.2 – Schéma bloc de la boucle de régulation PID de la température.....	37
Fig. III.3 – La réponse du système de 1 ^{er} ordre commandé par PID	37
Fig. III.4- Schéma équivalent d’un moteur à courant continu.....	38
Fig. III.5 – Schéma bloc du système de 2 ^{ème} ordre en boucle fermée avec et sans PID.....	39
Fig. III.6 – Réponses du système de 2 ^{ème} ordre avec et sans PID	39
Fig. III.7 – Réponse du système de 2 ^{ème} ordre commandé par PID avec perturbation.....	40
Fig. III.8 – Réponse du système de 2 ^{ème} ordre avec perturbation commandé par PID	40
Fig. III.9 – Schéma bloc d’un système de 3 ^{ème} ordre en boucle fermée avec et sans PID.....	41
Fig. III.10 – La réponse du système de 3 ^{ème} ordre commandé par PID.....	41
Fig. III.11 – Schéma bloc d’un système 3eme ordre en boucle fermée et avec un PID.....	42
Fig. III.12 – Réponse du système de 3eme ordre avec perturbation commandé par PID	42

Fig. III.13 – Création des fonctions d'appartenance des entrées et de sortie.	43
Fig. III.14 – Fuzzification de l'entrée du contrôleur flou.....	44
Fig. III.15 – Fuzzification de la sortie du contrôleur flou	44
Fig. III.16 – les règles floues.....	45
Fig. III.17 – l'agrégation des règles	45
Fig. III.18 – Surface des règles floues.....	46
Fig. III.19 – Schéma bloc de la boucle de régulation à contrôleur flou à une seule entrée.....	46
Fig. III.20 – Réponse du système du 1 ^{er} ordre commandé par un régulateur flou à une seule entrée	48
Fig. III.21 – Schéma bloc de la boucle de régulation à contrôleur flou à 2 entrées	48
Fig. III.22 – Réponse du système du 1 ^{er} ordre commandé par un régulateur flou à 2 entrées.....	49
Fig. III.23 – Schéma bloc de système de 2eme ordre commandé par le contrôleur flou.	50
Fig. III.25 – Schéma bloc de commande en régulation par le contrôleur flou	51
Fig. III.26 – La réponse du système de 2 ^{ème} ordre avec le contrôleur flou à 2 entrées avec perturbation.	52
Fig. III.27 – Schéma bloc de système de 3 ^{ème} ordre commandé par le contrôleur flou.....	53
Fig. III.28– La réponse du système de 3 ^{ème} ordre soumis au contrôleur flou.....	53
Fig. III.29– Schéma bloc de système de 3 ^{ème} ordre commandé en régulation.....	54
Fig. III.30 – La réponse du système de 3 ^{ème} ordre soumis au contrôleur flou avec perturbation.	54
Fig. III.31 – Schéma bloc de système du 1 ^{er} ordre commandé par le PID et le contrôleur flou	55
Fig. III.32 – La réponse du système du 1 ^{er} ordre.....	55
Fig. III.33 – Schéma bloc de système du 2 ^{ème} ordre commandé par le PID et le contrôleur flou	56
Fig. III. 34– La réponse du système du 2 ^{ème} ordre.....	56
Fig. III.35 – Schéma bloc de système du 3 ^{ème} ordre commandé par le PID et un contrôleur flou	57
Fig. III.36 – La réponse du système du 3 ^{ème} ordre.	57

LISTE DES TABLEAUX

Tab. I.1- Calcul des paramètres du contrôleur PID à partir de K_0 et T_0	17
Tab. II.1- Matrice d'inférence complète de Mac Vicar-Whelan pour deux variables d'entrée e et Δe .	29
Tab. III.1-Table des règles floues du contrôleur flou (333)	50

INTRODUCTION GENERALE

Introduction Générale

L'augmentation du prix des équipements industriels (actionneurs, capteurs,...), de la matière première et l'imposition de contraintes sévères sur les performances (rendement, normes de pollution et de consommation,...) imposent la recherche d'un fonctionnement optimal des systèmes. Les techniques de commande suscitent beaucoup d'attention depuis de nombreuses années car ces méthodes sont un bon moyen d'accroître les performances de processus, sur tous leurs domaines de fonctionnement. De plus, les capacités de traitement et la rapidité d'exécution des calculateurs actuels permettent l'implantation de stratégies de commande de plus en plus sophistiquées.

La démarche de l'automatique classique (approche algorithmique) consiste, dans un premier temps, à construire un modèle mathématique, à base d'équations différentielles ou récurrentes, du système à piloter. A partir de ce modèle, une commande est déterminée (P.I.D, commande par retour d'état, commande optimale, ...) afin d'amener ce système dans les états désirés, tout en respectant des critères de performances (durée des transitoires, minimisation de l'énergie apportée, ...). Mais, dans la première partie des années soixante, il est apparu de plus en plus douteux à Zadeh, automaticien américain possédant déjà une réputation internationale, que ce type d'approche soit toujours fiable pour les systèmes complexes (moteur thermique, processus chimiques, ...). En effet, l'obtention d'un modèle mathématique à la fois précis, pertinent et simple à exploiter s'avère parfois difficile.

Cette constatation a scellé les prémisses du développement des commandes à base de logique floue. Zadeh propose l'utilisation des ensembles flous pour décrire de manière « grossière » les relations entre les valeurs mal connues des variables décrivant l'état du système et les commandes appliquées. Au début des années 90, les japonais avaient poursuivi l'étude des contrôleurs à base de règles. Depuis, la commande floue suscite beaucoup d'intérêt.

L'objectif de ce travail est d'appliquer les deux méthodes de régulation qui sont la commande PID et la commande floue sur plusieurs systèmes avec et sans perturbation afin de pouvoir faire une étude comparative entre ces deux commandes et voir les avantages et les inconvénients de chacune d'entre elles. Pour ce faire, le mémoire est organisé comme suit :

Dans le premier chapitre, nous présenterons les notions de base des systèmes linéaires, les critères de qualité d'un asservissement (performance d'un système), puis nous décrivons les différents systèmes élémentaires : 1^{er}, 2^{ème} et 3^{ème} ordre. Ensuite nous introduisons le principe de la commande classique conventionnelle, les différents types des régulateurs P, PI, PID, leurs principe de fonctionnement, leurs avantages et inconvénients puis nous citons les différentes méthodes de synthèse d'un régulateur PID parmi les méthodes nous avons cité la méthode de Ziegler -Nichols .

Dans le deuxième chapitre, nous décrivons les principes de base de la logique floue. Nous introduisons la théorie des ensembles flous ainsi que les différents contrôleurs flous, et leurs caractéristiques. Nous présentons ensuite les deux types de contrôleurs flous les plus utilisés (Mamdani et Takagi-Sugeno), et nous terminons le chapitre par les avantages, les inconvénients et quelques domaines d'application de la logique floue.

Le troisième chapitre nous présentons notre travail de simulation où on a utilisé les deux régulateurs présentés précédemment (contrôleur PID, contrôleur flou) pour la régulation de quelques systèmes en 1^{er}, 2^{ème} et 3^{ème} ordre.

Nous terminerons ce mémoire par une conclusion générale dans laquelle nous donnerons les futures perspectives et les améliorations de ce travail.

CHAPITRE I
LES REGULATEURS CONVENTIONNELS PID

Chapitre I : les Régulateurs Conventionnels PID

I.1 Introduction

La première étape lorsqu' on veut analyser puis commander un système consiste à comprendre son bon fonctionnement, c-à-d, son comportement physique. Les méthodes d'étude des systèmes linéaires à partir de la notion de fonction de transfert et de ses représentations constituent un outil extrêmement puissant pour l'étude et la synthèse des systèmes asservis. On peut même dire que les méthodes d'optimisation ont porté cet outil à sa perfection, puisqu' elles permettent de déterminer le meilleur qu'on peut obtenir à partir des données du problème en restant dans le cadre de l'hypothèse linéaire.

L'obtention des performances exigées pour un système de commande, nécessite a joindre au système commandé des organes extérieurs appelés régulateurs ou contrôleurs.

Les régulations peuvent être linéaires ou non linéaires. Les régulateurs linéaires les plus utilisés sont : régulateur de type P (proportionnel), régulateur de type PI (proportionnel et intégral), régulateur de type PD (proportionnel et dérivé), régulateur de type PID (proportionnel, intégral, dérivé).

Du moment que les régulateurs continus PI, PD, PID, ont reçu une large application dans l'industrie. Le régulateur PID est certainement l'un des plus importants en raison de son universalité. Il est à noter que les régulateurs PID sont les plus répondu dans l'industrie et sont même utilisés, en combinaison avec des régulateurs flous.

Ce chapitre va introduire plusieurs notions sur les systèmes linéaires et non linéaires, notion de performances puis explique le principe de la commande classique ainsi que les différents types des régulateurs.

I.2 Notion de système

En automatique, on appelle système ou processus l'objet à étudier. La définition d'un système est liée aux grandeurs d'entrée et de sortie considérées. Il peut être défini comme un ensemble des éléments exerçant collectivement une fonction déterminée. Un système communique avec l'extérieur par l'intermédiaire de grandeurs, fonctions du temps, appelées signaux. En général, ces signaux (d'entrée et de sortie) ne sont pas de même nature. Les grandeurs d'entrée sont les grandeurs qui agissent sur le système. Il en existe deux types : Commandes (celles que l'on peut maîtriser) et perturbations (celles que l'on ne peut pas maîtriser).

Les signaux de sortie d'un système sont aussi appelés réponse du système. Un système possédant une seule entrée est dit mono variable, si de plus il possède une seule sortie, il est dit scalaire. Pour observer les grandeurs de sortie, on utilise des capteurs. C'est l'information de ces capteurs qui va permettre d'élaborer la commande.

I.2.1 La boucle d'asservissement [1]

Considérons le système de chauffage central d'un logement représenté figure I.1

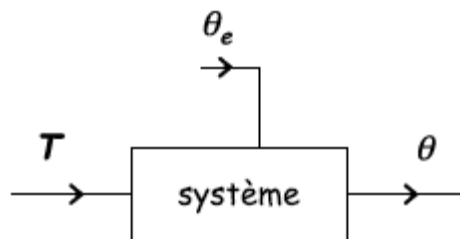


Fig. I.1 – Système de chauffage central

Avec : θ la température intérieure,

T la température de l'eau chaude envoyée dans les radiateurs,

θ_e la température extérieure (considérée comme une perturbation).

T est réglée par le chauffagiste pour obtenir une température de consigne donnée $\theta_c = 19^\circ\text{C}$.

Cependant, le réglage est à refaire à chaque variation de θ_e .

Une première tentative de réglage automatique en boucle ouverte est représentée figure I.2.

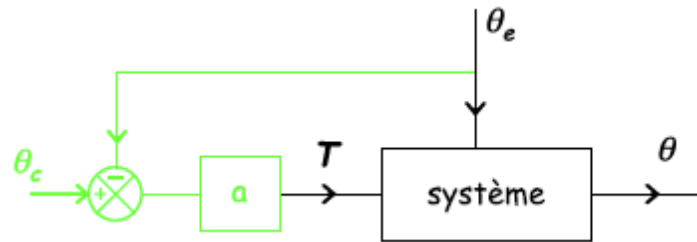


Fig. I.2 – Asservissement en boucle ouverte.

Une sonde est installée afin de mesurer θ_e et la mesure est soustraite à la température souhaitée θ_c (la consigne) pour élaborer la loi de commande fixant la température T de l'eau :

$$T = a.(\theta_c - \theta_e) \quad (\text{I.1})$$

Avec a est une constante réglable.

Ainsi, toute évolution de la température extérieure est prise en compte et la température de l'eau du circuit de chauffage ajustée en conséquence. Le savoir-faire du chauffagiste réside alors dans le choix de la constante a , l'ajustement pouvant être fait par essai-erreur.

Cette première approche présente une amélioration notable. Malheureusement, elle n'est pas encore optimale. En effet, lors d'une journée d'hiver ensoleillée (θ_e faible) T va être réglée à une valeur élevée, alors que le soleil entrant par les fenêtres va surchauffer le logement. Les habitants vont alors ouvrir les fenêtres entraînant un gaspillage d'énergie important.

La solution consiste à réaliser un asservissement en boucle fermée (figure I.3) du système de chauffage en exploitant une mesure de la température intérieure θ plutôt que d'essayer d'anticiper l'effet de la température extérieure θ_e sur le système.

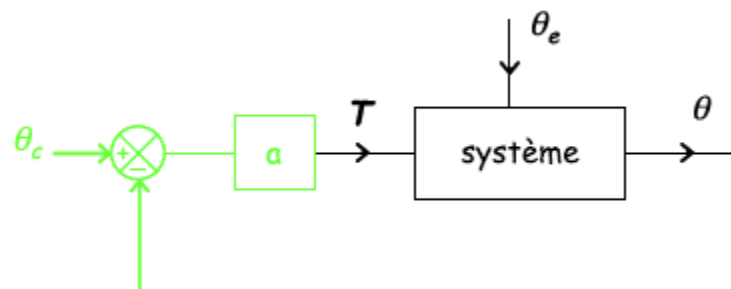


Fig. I.3 – Asservissement en boucle fermée d'un système de chauffage.

Le recours à une loi de commande proportionnelle est alors adapté par l'équation I.1. Réalisé ainsi, l'asservissement est à même de réagir aux variations de la température extérieure et aux changements de la consigne.

La figure I.4 donne une vue plus générale d'un asservissement en boucle fermée.

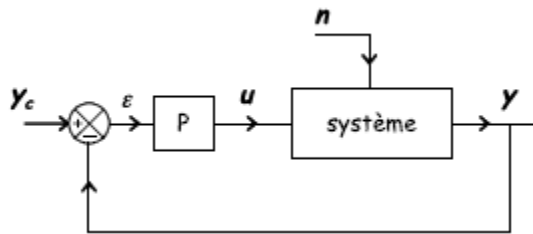


Fig. I.4 – Asservissement en boucle fermée.

Avec :

y_c est la consigne,

y est la sortie (ou image de la sortie obtenue par un capteur),

u est la commande ou action,

ε est l'erreur ou écart, tel que $\varepsilon = y_c - y$.

n est la perturbation extérieure,

P est le contrôleur.

La loi de commande étant $u = P(\varepsilon)$.

I.2.1.1 Qualité d'un asservissement et performances d'un système [2]

Les principales qualités d'un asservissement sont au nombre de trois : stabilité, précision, et rapidité.

I.2.1.1.1 Stabilité

Un système est dit stable si et seulement si à une entrée bornée $e(t)$ correspond une sortie bornée $s(t)$. Cette définition permet de qualifier la stabilité des systèmes forcés.

I.2.1.1.2 Rapidité

La sortie doit atteindre la valeur de consigne le plus rapidement possible. Le temps de réponse doit être le plus petit possible. Le temps de réponse à 5% d'un système est le temps

mis pour que sa sortie atteigne et reste dans l'intervalle [95% ; 105%] de la valeur finale stabilisée.

I.2.1.1.3 Précision

Il est naturel d'évaluer la précision d'un système régulé en comparant l'objectif atteint par rapport à celui exigé. La précision d'un système régulé se mesure donc à l'écart entre la consigne demandée et la mesure en régime permanent ; on parle alors de précision statique s'effectue en réalisant une variation rapide de consigne en amplitude et en mesurant la variation d'amplitude.

D'une façon générale, la synthèse d'un asservissement résulte d'un **compromis** stabilité – précision – rapidité.

I.3 Systèmes élémentaires

I.3.1 Notions de linéarité et non linéarité

On appelle système linéaire les systèmes physiques représentés par des équations différentielles linéaires à coefficients constants, et que l'hypothèse de linéarité équivaut au principe de superposition qui peut s'énoncer :

Si $s_1(t)$ et $s_2(t)$ sont les réponses respectives à deux entrées $e_1(t)$ et $e_2(t)$, la réponse à l'entrée $\alpha \cdot e_1(t) + \beta \cdot e_2(t)$ sera $\alpha \cdot s_1(t) + \beta \cdot s_2(t)$. Dans le cas d'un système linéaire à une entrée $e(t)$ et une sortie $s(t)$: si la réponse du système linéaire à $e(t)$ est $s(t)$, sa réponse à l'entrée $A \cdot e(t)$, A étant une constante, sera $A \cdot s(t)$.

Les systèmes non linéaires, par opposition aux systèmes linéaires, sont les systèmes physiques qui ne sont pas régis par des équations linéaires. Autrement dit, le principe de superposition ne s'applique pas.

Cependant, les systèmes physiques (réels) ne sont pas nécessairement linéaires. Il est néanmoins souvent possible de les étudier avec les outils classiques de l'automatique linéaire après avoir *linéarisé* leur comportement autour d'un point de repos.

I.3.2 Système du premier ordre [3]

Par définition, ces systèmes obéissent à une équation différentielle du premier ordre à coefficients constants :

$$x(t) = a \cdot \frac{dy(t)}{dt} + b \cdot y(t) \quad (\text{I.2})$$

Où $x(t)$ et $y(t)$ représentent respectivement l'entrée et la sortie du système il faut noter qu'un grand nombre de systèmes physiques peuvent être représentés par des modèles du premier ordre.

La forme générale de la fonction de transfert des systèmes du premier ordre

$$F(p) = \frac{Y(p)}{X(p)} = \frac{1}{ap + b} = \frac{1/b}{1 + \frac{a}{b}p} \quad (\text{I.3})$$

La forme standard d'une fonction de transfert d'un système du premier ordre

$$F(p) = \frac{K}{1 + \tau P} \quad (\text{I.4})$$

Le gain statique du système K s'exprime dans la même unité que le rapport $\frac{\Delta y(t)}{\Delta x(t)}$.

Pour un système stable (la sortie est constante en régime permanent). La constante de temps τ du système s'exprime en secondes et caractérise la vitesse (dynamique) de réaction d'un système, le gain statique représente la valeur prise par sa réponse à une entrée en échelon unitaire quand $t \rightarrow \infty$.

I.3.3 Système du 2^{ème} ordre [3]

Ces systèmes obéissent à une équation différentielle du deuxième ordre à coefficients constants :

$$x(t) = a \frac{d^2 y(t)}{dt^2} + b \frac{dy(t)}{dt} + cy(t) \quad (\text{I.5})$$

Où $x(t)$ et $y(t)$ représentent respectivement l'entrée et la sortie du système la forme générale de la fonction de transfert des systèmes du deuxième ordre.

$$G(p) = \frac{Y(p)}{X(p)} = \frac{1}{ap^2 + bp + c} = \frac{K\omega_0^2}{P^2 + 2\xi\omega_0 P + \omega_0^2} \quad (\text{I.6})$$

Avec :

K est appelé gain statique du système.

ω_0 est la pulsation propre non amortie du système (elle s'exprime en radians par seconde).

ξ est le facteur d'amortissement du système

Ces trois grandeurs sont suffisantes pour caractériser tout système du deuxième ordre.

I.3.4 Système d'ordre supérieur à 2

D'une façon générale, l'équation différentielle représentative d'un système linéaire supérieur à 2 peut s'écrire

$$a_0 \cdot y(t) + a_1 \cdot \frac{dy(t)}{dt} + \dots + a_n \cdot \frac{d^n y(t)}{dt^n} = b_0 \cdot u(t) + b_1 \cdot \frac{dy(t)}{dt} + b_m + \frac{d^m y(t)}{dt^m}$$

Avec a_i, b_i les coefficients constants réels

$n \geq m$ Pour les systèmes physiques réalisables (c'est-à-dire respectant le principe de causalité)

N est l'ordre du système.

On déduit l'expression de la fonction de transfert correspondante (condition initiales)

$$H(p) = \frac{N(p)}{D(p)} = \frac{b_m \cdot p^m + b_{m-1} \cdot p^{m-1} + \dots + b_1 \cdot p + b_0}{a_n \cdot p^n + a_{n-1} \cdot p^{n-1} + \dots + a_1 \cdot p + a_0}$$

Les racines du numérateur $N(p)$ sont les zéros de la fonction de transfert $H(p)$ et les racines du dénominateur $D(p)$ sont les pôles

Les coefficients étant réels, les n pôles (p_i à p_n) sont soit réel, soit complexes conjugués deux à deux

Ainsi le dénominateur peut s'écrire sous la forme

$$D(p) = a_n \cdot (p - p_1) \cdot (p - p_2) \cdot \dots \cdot (p - p_n)$$

D'où la possibilité d'exprimer $H(p)$ comme une somme d'éléments simples

$$H(p) = \sum_{i=1}^n \frac{A_j}{p - p_i} \quad \text{avec } A_j \text{ complexe}$$

En termes de stabilité, il suffit d'un seul pôle partie réelle positive pour entraîner l'instabilité de l'ensemble.

I.4 Les régulateurs PID

Le rôle du régulateur pour le système en boucle fermée est d'assurer que la réponse présente des caractéristiques dynamiques et stationnaires convenables [4].

On peut juger le contrôleur selon les critères suivants :

- Le contrôleur doit être capable de maintenir la variable commandée à sa consigne.
- Le système en boucle fermée doit être stable asymptotiquement et présenter une performance satisfaisante dans une large gamme de fréquences.
- L'influence des perturbations doit être minimale.
- Les réponses à des variations de consigne doivent être rapides et douces.
- Une action de commande excessive doit être évitée (la variable de commande $u(t)$ ne doit pas être trop sollicitée).
- Le système de commande doit être robuste : il doit être insensible aux variations du procédé et aux erreurs du modèle du procédé. Types de système de commande.

I.4.1 Système de commande en boucle ouverte [5]

Cette structure de commande est simple et peu coûteuse, mais malheureusement dans certaines applications où la précision est d'une grande importance et/ou les paramètres du système à commander sont variantes, elle n'est pas utilisée à cause de son imprécision.

On peut définir la commande en boucle ouverte comme un système où le signal de commande (d'entrée) est indépendant du signal réglé (sortie).

I.4.2 Système de Commande en Boucle Fermée

Une structure de commande en boucle fermée présentée par la Figure I.5 est définie comme un système où le signal de commande (d'entrée) dépend d'une façon ou d'une autre du signal de sortie. Les systèmes en boucle fermée sont couramment appelés des systèmes asservis. La structure de commande en boucle fermée (contre réaction) est capable de :

- Stabiliser un système instable en boucle ouverte.
- Compenser les perturbations externes.
- Compenser les incertitudes internes au processus lui-même.

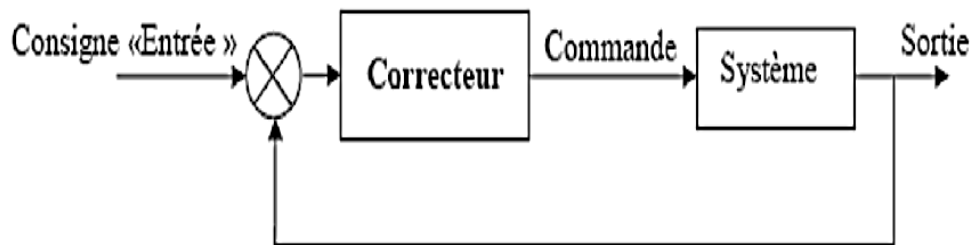


Fig. I.5 – commande en boucle fermée.

Le rôle du correcteur est d'éliminer l'écart, quelles que soient les perturbations, et d'amener le système à réagir le plus rapidement, quelles que soit les variations de l'entrée de référence ou des perturbations. Nous distinguons plusieurs types de correcteurs qui sont basés sur les actions : proportionnelle, intégrale et différentielle.

I.4.3 Types de contrôleurs [5]

Les régulateurs standards les plus utilisés dans l'industrie sont les régulateurs de la famille *PID* (proportionnel, intégral, dérivé), car ils permettent de régler à l'aide de ces trois paramètres les performances (amortissement, temps de réponse, dépassement...) d'une régulation d'un processus modélisé du premier ordre et du deuxième ordre.

La plupart des régulateurs comportent un algorithme identique à celui désigné par un "*PID*".

- **P** : Action proportionnelle.
- **I** : Action intégrale.
- **D** : Action dérivée

L'adaptation d'un régulateur aux différents systèmes s'effectue par le réglage des coefficients (paramètres) de l'algorithme.

- Paramètre K_p pour le réglage de l'action proportionnelle.
- Paramètre K_i pour le réglage de l'action intégrale.
- Paramètre K_d pour le réglage de l'action dérivée.

I.4.3.1 Correcteur proportionnel (P) [5]

Ce correcteur élémentaire est le correcteur de base, il agit principalement sur le gain du système asservi, il permet donc d'améliorer notablement la précision. Dans le cas d'un correcteur proportionnel, la loi de commande corrigée $u(t)$ est proportionnelle à l'écart $e(t)$

$$u(t) = K \cdot e(t) \quad (\text{I.7})$$

$$c(p) = \frac{U(p)}{E(p)} = K(p) \quad (\text{I.8})$$

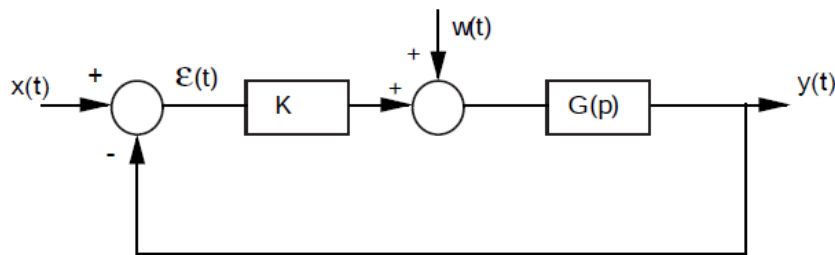


Fig. I.6 – Commande proportionnelle P.

L'effet d'une augmentation du gain entraîne une diminution de l'erreur statique, et rend le système plus rapide mais engendre l'instabilité du système.

I.4.3.2 Correcteur proportionnel – intégrale (PI) [5]

L'action proportionnelle est une fonction qui fait varier le signal de sortie du régulateur proportionnellement à l'écart entre la mesure et la consigne. Le correcteur intégral est fréquemment combiné au correcteur proportionnel pour améliorer le régime permanent et le régime transitoire. Cette combinaison est appelée correcteur proportionnel-intégral.

Le coefficient de proportionnalité, appelé gain du régulateur, est obtenu par le réglage du paramètre du régulateur.

L'équation dans le domaine temporel de ce correcteur est écrite par l'expression :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau \quad (\text{I.9})$$

Où $e(t)$, $u(t)$, K_p et K_i désignent respectivement l'erreur à l'instant t , la commande générée et les gains du correcteur.

La fonction de transfert du contrôleur correspondante est donnée par :

$$C(p) = K_p + \frac{K_i}{p} = K_p \left(1 + \frac{1}{T_i p}\right) \quad (\text{I.10})$$

$$\text{Avec : } T_i = \frac{K_p}{K_i}$$

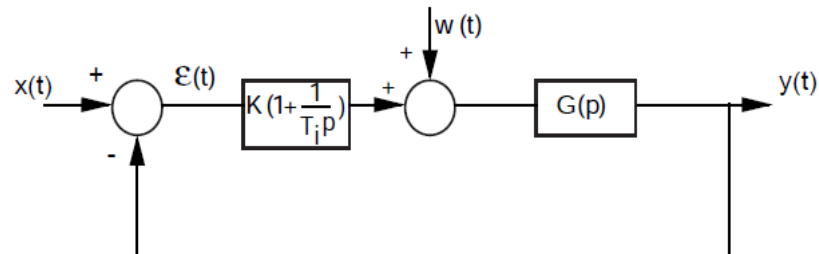


Fig. I.7 – Commande proportionnelle-intégrale PI.

L'introduction d'un correcteur PI permet d'améliorer la précision et de rejeter les perturbations de type échelon. Par contre, ce type de correcteur possède certaines limitations sur l'amélioration de la rapidité et peut même introduire une instabilité du système en boucle fermée [3].

I.4.3.3 Correcteur proportionnel – dérivée (PD) [5]

L'inconvénient majeur du correcteur dérivé est son insensibilité aux variations lentes de l'erreur. Pour cette raison, ce type de correcteur n'est jamais utilisé seul. La solution permettant de minimiser cet inconvénient est de le combiner au correcteur proportionnel [6].

Nous représentons le correcteur PD dans le domaine du temps par l'équation suivante :

$$u(t) = K_p e(t) + K_d \frac{de(t)}{dt} \quad (\text{I.11})$$

K_d : Gain de l'action dérivée.

La fonction de transfert correspondante est :

$$C(p) = K_p + K_d p = K_p (1 + T_d p) \quad (\text{I.12})$$

$$\text{Avec : } T_d = \frac{K_d}{K_p}$$

Ce type de correcteur possède la structure suivante :

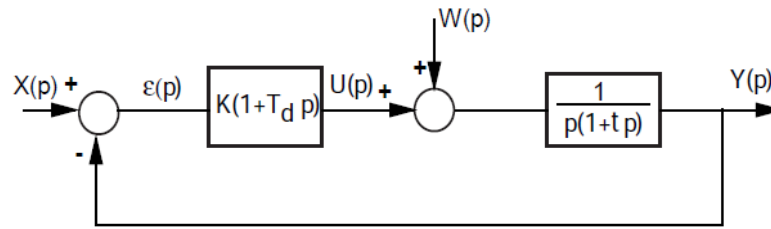


Fig. I.8 – Commande proportionnelle-dérivée PD.

Cette action n'a pas d'effet que lorsqu'il y a variation de l'erreur, elle n'a aucun effet sur le fonctionnement en régime permanent. Les correcteurs PD permet donc d'améliorer le régime transitoire [3].

I.4.3.4 Correcteur proportionnel - intégrale – dérivée [5]

Les termes proportionnel et intégral peuvent amener un dépassement de la consigne et des oscillations. Cela implique pour le moteur par exemple des inversions de polarité, ce qui est loin d'être idéal. Pour limiter ce phénomène indésirable, on introduit un troisième élément qui est le terme dérivé. Son action va dépendre du signe et de la vitesse de variation de l'erreur, et sera opposée à l'action proportionnelle. Elle devient prépondérante aux abords de la valeur demandée lorsque l'erreur devient faible, que l'action du terme proportionnel faiblit et que l'intégrale varie peu : elle freine alors le système, limitant le dépassement et diminuant le temps de stabilisation [7].

Le schéma général de la boucle fermée est donné par :

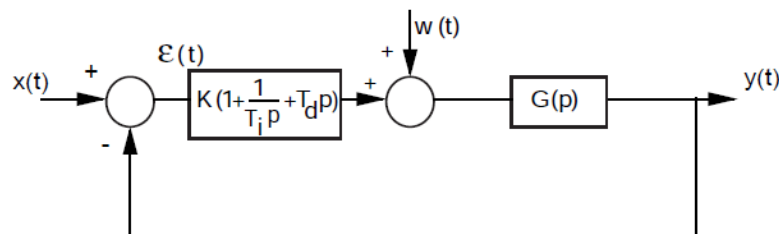


Fig. I.9 – Commande PID

L'intérêt du correcteur PID est d'intégrer les effets positifs des trois correcteurs précédents. La détermination des coefficients K_p , K_i , K_d du correcteur PID permet d'améliorer à la fois la précision (K_p et K_d), la stabilité (K_d) et la rapidité (K_p , K_d).

Le réglage d'un PID est en général assez complexe, des méthodes pratiques de réglages permettent d'obtenir des bons résultats [8]. Il est régi par la relation suivante :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (\text{I.13})$$

Sa fonction de transfert est donnée par :

$$C(s) = K_p + \frac{K_i}{p} + K_d p = K_p \left(1 + T_d p + \frac{1}{T_i p} \right) \quad (\text{I.14})$$

En général, on a les qualités suivantes du correcteur PID

- L'action proportionnelle est souvent utilisée pour la rapidité du système bouclé ;
- L'action intégrale est employée pour améliorer l'erreur en régime permanent ;
- L'action dérivée assure l'amélioration de la stabilité

I.4.4 Les différentes structures du PID [9]

Différentes possibilités d'associations des modules P, I et D existent. Ces structures sont fonctionnellement équivalentes, et il est facile de convertir les coefficients utilisés dans l'une pour obtenir ceux d'une autre. Les trois configurations les plus utilisées sont :

I.4.4.1 La structure parallèle

La loi de commande est de la forme :

$$C(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (\text{I.15})$$

Soit d'après la transformée de Laplace

$$C(p) = K_p e(p) + K_i \cdot \frac{e(p)}{p} + K_d \cdot p \cdot e(p) \quad (\text{I.16})$$

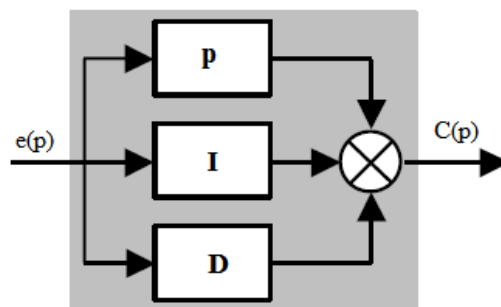


Fig. I.10 – La structure parallèle

I.4.4.2 La structure mixte

Soit d'après la transformée de Laplace, La loi de commande est de la forme :

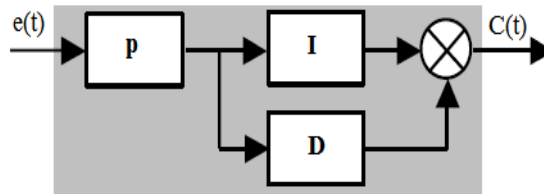


Fig. I.11 – La structure mixte

$$C(p) = K_p e(p) \cdot [K_i \cdot \frac{1}{p} + K_d \cdot p] \quad (\text{I.17})$$

I.4.4.3 La structure série

La transformée de Laplace de la loi de commande est de la forme :

$$C(p) = e(p) \cdot K_p \cdot [K_i \cdot \frac{1}{p}] \cdot [K_d \cdot p] \quad (\text{I.18})$$



Fig. I.12 – La structure série.

I.5 Méthodes de synthèse du régulateur PID [10]

Il y a plusieurs de méthodes qui permettent de calculer les paramètres du régulateur PID comme :

- Méthode de Halman
- Méthode de Cohen-Coon
- Méthode de Chien-Hrones-Reswick
- Méthode Ziegler –Nichols

On a choisi la méthode de Ziegler –Nichols parce que c'est une méthode très utilisée (pratique) qui permet d'évaluer correctement les paramètres du régulateur.

I.5.1 Méthode de Ziegler -Nichols

Plusieurs méthodes expérimentales ont été développées pour déterminer ces coefficients d'un correcteur PID. En 1942, Ziegler et Nichols ont proposé deux approches expérimentales destinées à ajuster rapidement les paramètres des régulateurs P, PI et PID. La première nécessite l'enregistrement de la réponse indicielle du système à régler seul, alors que la deuxième demande d'amener le système en boucle fermée à sa limite de stabilité. Il est important de souligner que ces méthodes ne s'appliquent en général qu'à des systèmes sans comportement oscillant et dont le déphasage en hautes fréquences dépasse -180 degrés. Ces systèmes possèdent souvent un retard pur et/ou plusieurs constantes de temps. On les rencontre surtout dans les processus physico-chimiques tels que les réglages de température, de niveau, de pression, etc [11].

Cette méthode est basée sur la connaissance du point critique du diagramme de nyquist du procédé. Les paramètres du régulateur PID en fonction des caractéristiques K_0 et T_0 sont donnés.

Dans le tableau 2 ou K_0 et T_0 sont respectivement le gain qui amène le system en oscillation limite et la période des oscillations. La Méthode de Ziegler –Nichols a été développée à partir de simulation sur un grand nombre de cas. Le critère de synthèse est d'obtenir un rapport de décroissance $d=1/4$, pour le rejet des perturbations, ce qui représente un coefficient d'amortissement z de 0.22. Ces valeurs sont satisfaisantes pour un bon rejet des perturbations, mais elles sont trop faibles pour les réponses aux changements de consigne.

Régulateur	Paramètres		
	K_p	K_i	K_d
P	K_0	-	-
PI	$K_0/2.2$	K_p/T_0	-
PID	$0.6 K_0$	$2 K_p/T_0$	$K_p T_0 / 8$

Tabl. I.1- Calcul des paramètres du contrôleur PID à partir de K_0 et T_0

I.6 Conclusion

Dans ce chapitre, nous avons présenté les notions de base de la commande conventionnelle, la structure de réglage classique avec une définition des régulateurs les plus utilisés (P, PI, PD et PID) Ainsi que leur principe de fonctionnement détaillée. Après a cité quelque méthodes de conception du régulateur PID et a présenté brièvement la plus utilisés (la méthode de Ziegler et Nichols).

CHAPITRE II
LA COMMANDE FLOUE

Chapitre II : La Commande Floue

II.1 Introduction

La logique floue est développée par Lotfi A. Zadeh en 1965 à partir de sa théorie des sous-ensembles flous [12]. Les sous-ensembles flous sont une manière mathématique de représenter l'imprécision de la langue naturelle, ils peuvent être considérés comme une généralisation de la théorie des ensembles classiques [12-15]. La logique floue est aussi appelée "logique linguistique" car ses valeurs de vérité sont des mots du langage courant : "plutôt vrai, presque faux, loin, si loin, près de, grand, petit...". La logique floue a pour objectif l'étude de la représentation des connaissances imprécises, des raisonnements approchés [16] et elle cherche à modéliser les notions vagues du langage naturel pour pallier l'inadéquation de la théorie des ensembles classiques dans ce domaine [17].

Dans ce chapitre, nous discutons sur la terminologie de base de la logique floue, et en donnant un rappel sur les concepts de base de la logique floue, et les différents types des contrôleurs flous.

II.2 Théorie des sous-ensembles

Dans la théorie des ensembles classiques, l'appartenance d'un élément à un sous-ensemble est booléenne (un ou zéro). Les sous-ensembles flous permettent en revanche de connaître le degré d'appartenance d'un élément au sous-ensemble. Un sous-ensemble flou A d'un univers du discours U est caractérisé par une fonction d'appartenance [12].

$$\mu_A: U \rightarrow [0,1] \quad (\text{II.1})$$

Où $\mu_A(x)$ est le niveau ou degré d'appartenance d'un élément de l'univers de discours U dans le sous-ensemble flou. On peut définir aussi un sous-ensemble flou A dans l'univers du discours U comme suit [12] :

$$\bar{A} = \{(x, \mu_A(x)) \mid x \in U\} \quad (\text{II.2})$$

En attribuant les éléments x de X un degré d'appartenance, on remarque les trois cas suivants

$$\begin{cases} \mu_A(x) = 0 \\ 0 < \mu_A(x) < 1 \\ \mu_A(x) = 1 \end{cases} \quad (\text{II.3})$$

Où $\mu_A(x) = 0$ si x n'appartient pas à A ; $0 < \mu_A(x) < 1$ si x appartient partiellement à A ; Et $\mu_A(x) = 1$ si x appartient entièrement à A . La fonction d'appartenance $\mu_A(x)$ inclut ou exclut donc à ses extrémités, tout élément x au sous-ensemble A , mais entre les valeurs extrêmes du degré d'appartenance [18].

II.2.1 Univers du discours

L'ensemble de références d'un mot du langage naturel s'appelle l'univers du discours. On peut le définir comme un ensemble de terme qui évoque le même concept mais a degrés différents. Il peut être fini ou non.

II.2.2 Variables linguistiques

Dans la théorie de logique floue, les concepts des systèmes sont représentés par des variables linguistiques. Une variable linguistique est une variable qui représente un état dans le système dont les valeurs sont des mots ou des phrases utilisées couramment dans une langue naturelle, chaque variable est caractérisée par un ensemble tel que :

$$\{X, T(x), U, G, M\} \quad (\text{II.4})$$

Dont : x est le nom de la variable.

$T(x)$ est l'ensemble des valeurs linguistiques que peut prendre x .

U est l'univers du discours associé avec la valeur de base.

G est la règle syntaxique pour générer les valeurs linguistiques de x .

M est la règle sémantique pour associer un sens à chaque valeur linguistique.

II.2.3 Les règles linguistiques

L'idée principale des systèmes basés sur la logique floue, est d'exprimer la connaissance humaine sous forme des règles linguistiques de forme **si...Alors...** chaque règle a deux parties :

- Partie antécédente (prémisse ou condition), exprimée par **si**
- Partie conséquence (conclusion) exprimée par **Alors**

La partie antécédente est la description de l'état du système. La partie conséquence exprime l'action que l'opérateur doit exécuter. Chaque règle floue est basée sur l'implication floue.

Il y a plusieurs formes de règles floues, la forme générale est :

Si (un ensemble de conditions est satisfait) **Alors** (un ensemble de conséquences peut être exécuté) [19].

II.2.4 Support

Il décrit l'ensemble des éléments qui sont partiellement dans A

$$S(A) = \{x \in U / \mu_A(x) > 0\} \quad (\text{II.5})$$

- L'élément x de X tel que $\mu_A(x) = 0.5$ est appelé « **point de commutation** ».
- Un ensemble flou dont le support est un singleton dans X avec $\mu_A(x)=1$ est appelé « **Singleton flou** » [19].

II.2.5 Hauteur

C'est la valeur maximale de la fonction d'appartenance, elle est donnée par la relation suivante :

$$h(A) = \max_{x \in X} \{\mu_A(x)\} \quad (\text{II.6})$$

Un sous ensemble flou est dit normalisé sa hauteur vaut 1 [20].

II.2.6 Noyau

On peut définir le noyau par l'équation suivante : C'est l'ensemble des éléments booléens qui appartiennent totalement à l'ensemble flou A , pour les fonctions de type triangulaire, le noyau est un singleton qui est appelé aussi valeur modale [19].

$$\eta(A) = \{x \in X / \mu_A(x) = 1\} \quad (\text{II.7})$$

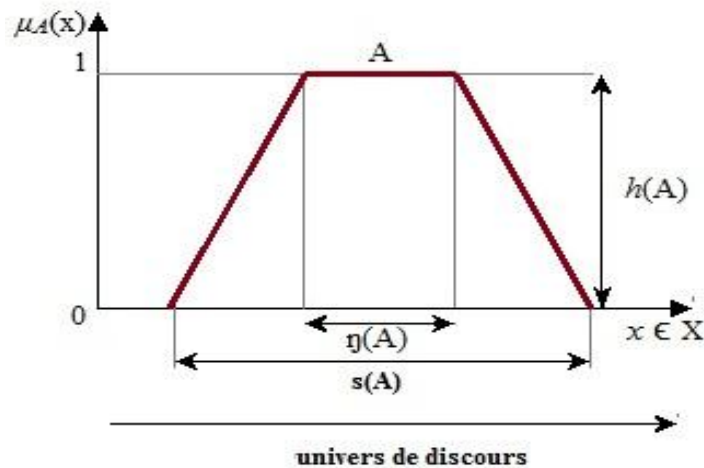


Fig. II.1 – Caractéristiques d'un sous-ensemble flou A [21].

II.2.7 Fonction d'appartenance

Soit un ensemble E et un sous-ensemble A de E ($A \subset E$), et x un élément de A appartenant à A ($x \in A$). Pour illustrer cette caractéristique, on utilise la fonction d'appartenance $\mu_A(x)$ compris entre 0 et 1, qui représente le degré d'appartenance de x à l'ensemble flou A . Le plus souvent, la fonction d'appartenance est déterminée par l'une des fonctions suivantes :

II.2.7.1 Fonction triangulaire

Elle est définie par trois paramètres (a, b, c) qui déterminent les coordonnées des trois sommets (figure 1.2-a)

$$\mu(x) = \max \left(\min \left(\frac{x-a}{b-a}, \frac{c-x}{c-b} \right), 0 \right) \quad (\text{II.8})$$

II.2.7.2 Fonction trapézoïdale

Elle est définie par la relation suivante :

$$\mu(x) = \max \left(\min \left(\frac{x-a}{b-a}, 1, \frac{c-x}{c-b} \right), 0 \right) \quad (\text{II.9})$$

II.2.7.3 Fonction gaussienne

Elle est définie par deux paramètres (σ, m)

$$\mu(x) = \exp \left(-\frac{(x-m)^2}{2\sigma^2} \right) \quad (\text{II.10})$$

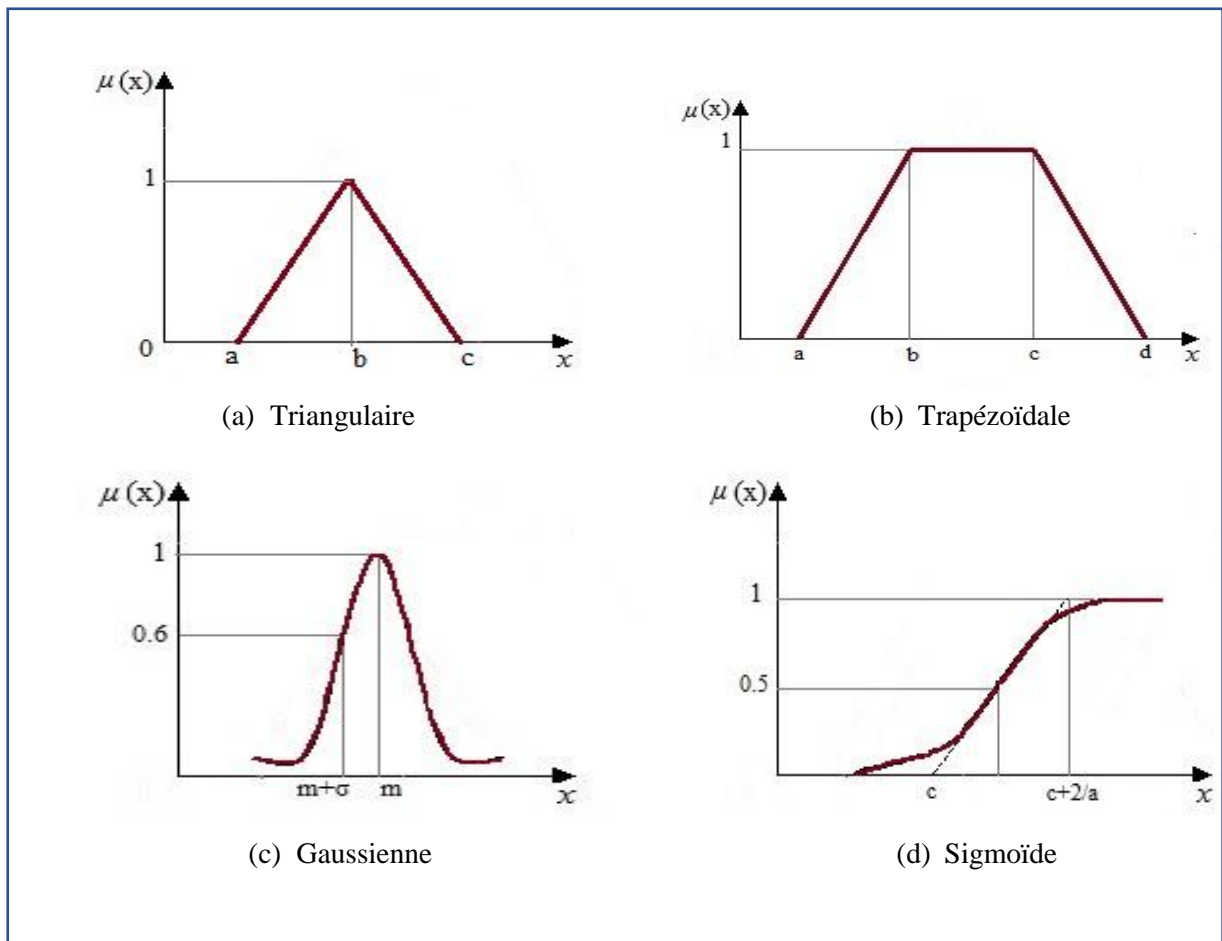


Fig. II.2 – fonction appartenance [21].

II.2.7.4 Fonction sigmoïde

Elle est définie par deux paramètres (a, c) :

$$\mu(x) = \frac{1}{1 + \exp(-a(x-c))} \quad (\text{II.11})$$

II.2.8 Les opérations sur les ensembles flous

Le concept de sous-ensemble flou de l'ensemble X étant une généralisation de la notion de sous-ensemble classique de X , ces opérations sont choisies de façon à être équivalentes aux opérations classiques de la théorie des ensembles lorsque les fonctions d'appartenance ne prennent que les valeurs 0 ou 1. Étant donné deux sous-ensembles flous A et B de X .

II.2.8.1 Egalité

On dit que les deux sous-ensembles flous A et B sont égaux si leurs fonctions d'appartenance prennent la même valeur pour tout élément de X .

$$A = B \quad \text{si et seulement si} \quad \mu_A(x) = \mu_B(x); \forall x \in E \quad (\text{II.12})$$

II.2.8.2 Union

L'union de deux sous-ensembles flous A et B de X est le sous-ensemble flou que l'on note $A \cup B$, tel que :

$$\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)]; \forall x \in E \quad (\text{II.13})$$

II.2.8.3 Intersection

L'intersection de deux sous-ensembles flous A et B de X est le sous-ensemble flou C , que l'on note $A \cap B$, tel que :

$$\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)]; \forall x \in E \quad (\text{II.14})$$

II.2.8.4 Complément

Le complément \bar{A} d'un sous-ensemble flou A de X est défini comme le sous ensemble flou de X de la fonction d'appartenance. Il est donné par l'équation suivante :

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x); \forall x \in E \quad (\text{II.15})$$

II.2.8.5 Inclusion

On dit que A est inclus dans B , et on note $A \subseteq B$, si leurs fonctions d'appartenance sont telles que

$$A \subseteq B \quad \text{si et seulement si} \quad \mu_A(x) \leq \mu_B(x) \quad (\text{II.16})$$

II.3 La commande floue

La commande floue est un domaine d'application de la théorie des ensembles flous qui a été proposé par L. Zadeh et développé après par plusieurs chercheurs comme (Mamdani, Assilian, etc). Son but est de traiter des problèmes de commande de processus (machines-outils, procédés industriels, véhicules sans pilote...etc.), c'est-à-dire de gérer un processus en fonction d'une consigne donnée par action sur les variables qui décrivent le processus, mais son approche est différente de l'automatique. Elle se sert des connaissances des experts ou d'opérateurs qualifiés travaillant sur le processus [8]

II.3.1 Description générale d'un contrôleur flou

Un contrôleur flou est un système à base de connaissances, utilisant un raisonnement en profondeur limité, dans une procédure de chaînage avant règles (activation des règles par les prémisses). On peut distinguer plusieurs étapes dans le traitement des règles. Un schéma représentatif peut être le suivant :

Les variables caractéristiques du système à commander et les consignes définissent les variables d'entrée du contrôleur flou. Les variables caractéristiques sont en général, les grandeurs de sortie du processus et d'autres mesures déterminantes pour saisir l'évolution

dynamique du processus. Les variables de sortie du contrôleur flou sont les commandes à appliquer au processus.

La base de connaissances est composée d'une base de données et d'une base de règles. La base de données regroupe :

- Les ensembles flous associés aux variables d'entrée et de sortie du contrôleur flou,
- Les facteurs d'échelle (gains) en entrée (normalisation) et en sortie (dénormalisation).

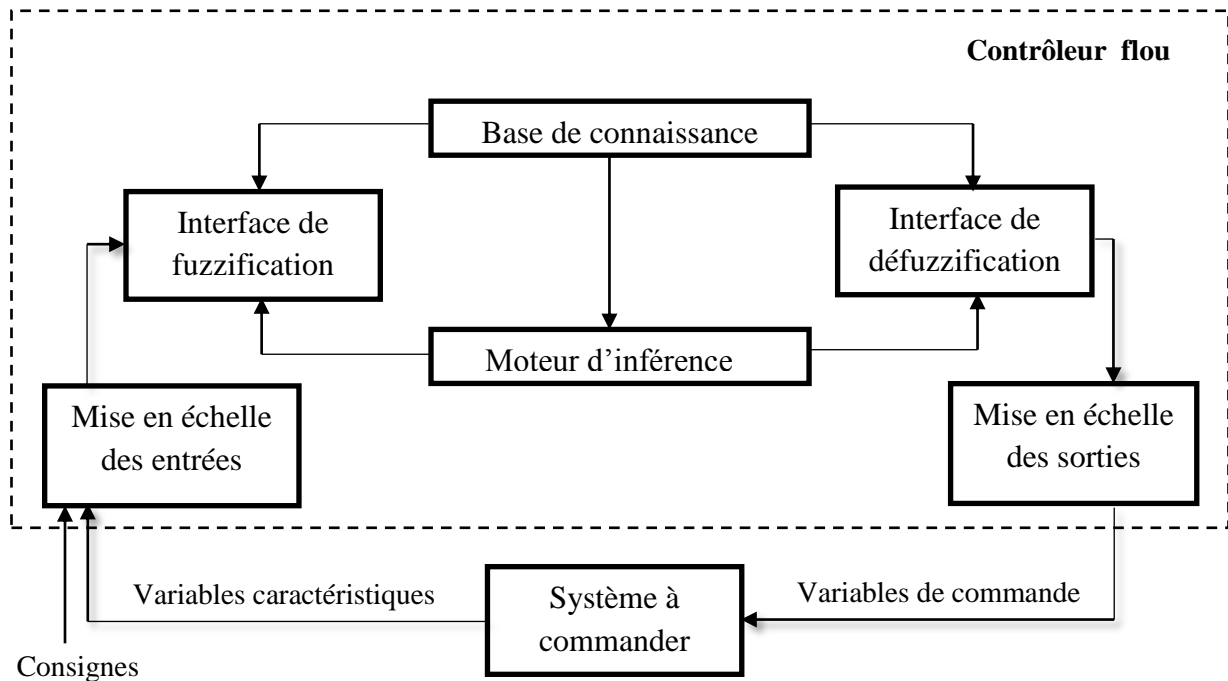


Fig. II.3 – structure de base d'un contrôleur flou [21].

II.3.2 Interface de fuzzification

C'est l'étape de transformation des variables physiques en valeurs floues au travers de variables linguistiques. La fuzzification est la procédure qui interface l'état non flou du système au domaine flou, dans cette étape la variable non floue x est conditionnée par les fonctions d'appartenance pour obtenir les valeurs floues $\mu_A(x)$ Correspondantes. Les systèmes à logique floue traitent des variables floues et fournissent des résultats et des variables de sorties elles-mêmes floues, la fuzzification est l'étape qui consiste en la quantification floue des valeurs réelles d'une variable (la figure II.4).

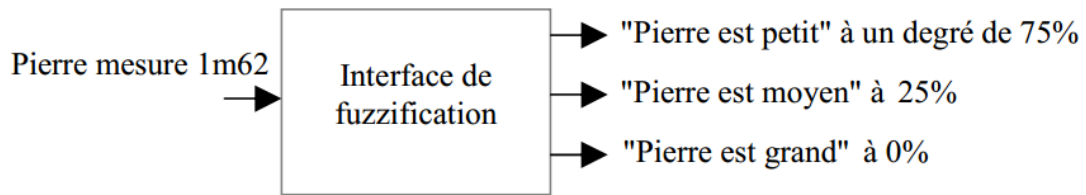


Fig. II.4 – structure de base d'un contrôleur flou

☞ Mais Comment fuzzifier ?

Pour fuzzifier, il faut donner.

1. L'univers du discours : Plage de variations possibles de l'entrée considérée.
2. Une partition en classes floues de cet univers.
3. Les fonctions d'appartenances de chacune de ces classes.

Il faut fuzzifier les entrées et les sorties du processus flou. La fuzzification des variables est une phase délicate du processus mis en œuvre par la logique floue. Elle est souvent réalisée de manière itérative et requiert de l'expérience.

Exemple : Un exemple de fuzzification continue est illustré sur la figure II.5, pour une seule variable x , avec des fonctions d'appartenance trapézoïdales. Ainsi si $x_1 = 0.2$, après fuzzification le sous-ensemble flou $\mu(x_1) = (\mu_N, \mu_Z, \mu_P) = (0, 0.8, 0.2)$.

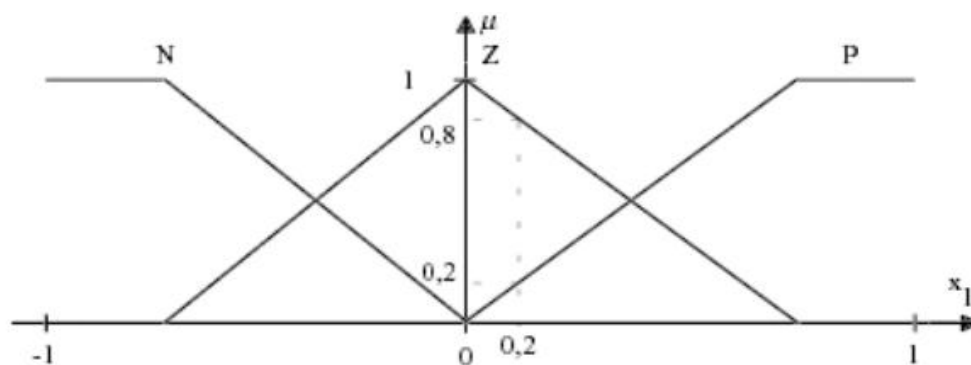


Fig. II.5 – Fuzzification continue avec trois fonctions d'appartenance

Cependant la forme de ces sous-ensembles est définie par :

- Les fonctions d'appartenance

- Les positions des fonctions d'appartenance
- Les chevauchements des fonctions d'appartenance

Les sous-ensembles d'entrées doivent se chevaucher partiellement afin qu'il n'y ait pas des zones indéterminées, tandis que pour ceux de sorties, cette précaution n'est pas nécessaire. Eviter d'imbriquer plus de deux sous-ensembles, sinon il y'a des problèmes d'instabilité.

II.3.3 Base de connaissances [22]

La conception d'une base de connaissances représente la phase dans la conception des systèmes experts. Elle comprend la base de données et la base des règles floues

- **La base de données**

Contient la définition des ensembles flous, les facteurs d'échelle pour la normalisation des ensembles de référence et la partition de l'espace flou d'entrée et sortie.

- **La base des règles floues**

La base de règle contient des règles de la forme :

« Si X_1 est A_1 et X_2 est A_2 ...et X_n est A_n Alors Y est B »

Ou X_1, X_2, X_n, Y sont des grandeurs physiques caractéristiques du système et du problème de commande. A_1, A_2, \dots, A_n et B sont des labels linguistiques.

Suivant la nature de B on parlera de :

- Règles à conclusion symbolique (contrôleur de type **Mamdani**) : B est une valeur linguistique. *Exemple :*

Si l'erreur est « Négatif Moyen » et la variation de l'erreur est « Positif Petit » **Alors** la commande est « Négatif Petit ».

- Règles à conclusion algébrique (contrôleur de **Sugeno**) : B est une valeur numérique (singleton) ou une équation mathématique bien précise (non floue). *Exemple :*

Si l'erreur est « Négatif Moyen » et la variation de l'erreur est « Positif Petit » Alors la commande est -0.3. Lorsque B est une valeur numérique on parle de règles de Takagi-Sugeno « d'ordre zéro », sinon, de règles à conclusion polynomiale [19].

Mac Vicar-Whelan ont observé que les bases de règles proposées par King et Mamdani étaient incomplètes [21]. Ils ont fait une analyse détaillée sur ces bases de règles et proposé

une matrice de règles (voir tableau II.1) qui possède deux entrées, l'erreur et sa variation, en se basant sur les deux principes suivants [23]:

- Si la sortie est égale à la valeur désirée et la variation de l'erreur est nulle, la Commande sera maintenue constante.
- Si la sortie diverge de la valeur désirée, l'action sera dépendante du signe de la Valeur de l'erreur et de sa variation.
- Si les conditions sont telles que l'erreur peut être corrigée par elle-même, alors la Commande sera maintenue. Dans le cas contraire, la commande sera changée pour avoir des résultats satisfaisants.

		e				
		NG	NP	ZE	PP	PG
Δe	NG	NG	NG	NP	NP	ZE
	NP	NG	NP	NP	ZE	PP
	ZE	NP	NP	ZE	PP	PP
	PP	NP	ZE	PP	PP	PG
	PG	ZE	PP	PP	PG	PG

Tab. II.1- Matrice d'inférence complète de Mac Vicar-Whelan pour deux variables d'entrée e et Δe .

II.3.4 Moteur d'inférence floue

C'est un mécanisme de décision. Il permet à partir d'un fait observé de la base des règles floues une décision en exploitant le raisonnement approximatif. Dans les inférences de régulateur par logique floue interviennent les opérateurs **ET** et **OU**. L'opérateur **ET** s'applique aux variables à l'intérieur d'une règle tandis que l'opérateur **OU** lie les différentes règles. Il existe plusieurs méthodes pour réaliser ces opérateurs dans une inférence et qui s'appliquent aux fonctions d'appartenance à savoir [22] :

- Méthode d'inférence max-min (Mamdani)
- Méthode d'inférence max-produit (Larsen)
- Méthode d'inférence somme-produit (Sugeno)

II.3.4.1 Méthode d'inférence max-min

La méthode d'inférence max-min réalise au niveau de la condition l'opérateur OU par la formation du maximum et l'opérateur ET par la formation du minimum ainsi que l'implication Alors est réalisée par la formation du minimum [22].

Pour chaque règle, on obtient la fonction d'appartenance partielle par la relation

$$\mu_{Ri}(x_R) = \min[\mu_{ci}, \mu_{oi}(x_R)]; i = 1, 2, \dots, m. \quad (\text{II.17})$$

Où $\mu_{oi}(x_R)$ est la fonction d'appartenance liée à l'opération imposée par la règle Ri , μ_{ci} est le facteur d'appartenance. La fonction d'appartenance résultante est alors donnée par

$$\mu_{Res}(x_R) = \max[\mu_{R1}(x_R), \mu_{R2}(x_R), \dots, \mu_{Rm}(x_R)] \quad (\text{II.18})$$

II.3.4.2 Méthode d'inférence max-produit

La méthode d'inférence max-produit réalise en général, au niveau de la condition, l'opérateur OU par la formation du maximum et l'opérateur ET par la formation du minimum, et l'implication Alors est réalisée par la formation du produit.

Pour chaque règle, on obtient la fonction d'appartenance partielle par la relation.

$$\mu_{Ri}(x_R) = \mu_{ci} \times \mu_{oi}(x_R); i = 1, 2, \dots, m \quad (\text{II.19})$$

Pour la fonction d'appartenance résultante, on obtient :

$$\mu_{Res}(x_R) = \max[\mu_{R1}(x_R), \mu_{R2}(x_R), \dots, \mu_{Rm}(x_R)] \quad (\text{II.20})$$

II.3.4.3 Méthode d'inférence somme-produit

Par opposition aux méthodes d'inférence précédentes, la Méthode d'inférence somme-produit réalise, au niveau de la condition, l'opération OU par la formation de la somme, plus précisément par la valeur moyenne, tandis que l'opération ET est réalisée par la formation du produit et l'implication Alors est réalisée par la formation du produit [22]

En toute généralité, on obtient la fonction d'appartenance partielle $\mu_{Ri}(x_R)$ de chaque règle par la relation

$$\mu_{Ri}(x_R) = \mu_{ci} \times \mu_{oi}(x_R); i=1,2 \dots \dots, m \quad (\text{II.21})$$

La fonction d'appartenance résultante est donnée par :

$$\mu_{Res}(x_R) = \sum[\mu_{R1}(x_R), \mu_{R2}(x_R), \dots \mu_{Rm}(x_R)]/m \quad (\text{II.22})$$

Où m est le nombre de règles intervenant dans l'inférence.

II.3.5 Interface de défuzzification

La stratégie de commande utilisée est décrite par un ensemble de déclaration linguistique ou règles. Puisque le processus exige à ses entrées des valeurs non floues, l'ensemble flou de sortie du régulateur doit être défuzzifié. La défuzzification est l'opération qui consiste à extraire d'un ensemble flou une valeur numérique considérée comme valeur significative. Dans la commande en temps réel, un critère de choix de la commande de défuzzification est la simplicité de calcul [22].

Les méthodes les plus utilisées sont [21]:

- Méthode de centre de gravité.
- Méthode du maximum.
- Méthode de la moyenne des maximas

➤ *Méthode de centre de gravité*

La commande résultante u_r représente le centre de gravité de l'ensemble flou inféré :

Dans le cas discret

$$u_r = \frac{\sum_{i=1}^n u_i \mu_{Res}(u_i)}{\sum_{i=1}^n \mu_{Res}(u_i)} \quad (\text{II.23})$$

n : Le nombre de niveaux de quantification de la sortie du contrôleur flou.

Dans le cas continu :

$$u_r = \frac{\int u \mu_{Res}(u) du}{\int \mu_{Res}(u) du} \quad (II.24)$$

➤ **Méthode du maximum**

Cette méthode, s'applique uniquement dans le cas où la fonction d'appartenance associée à l'ensemble de sortie n'admet qu'un seul maximum. On choisit comme sortie l'abscisse u_r correspondant à ce maximum :

$$\mu_A(u_r) = \max (\mu_A(u)) \quad (II.25)$$

➤ **Méthode de la moyenne des maximas**

Dans cette méthode, la valeur de sortie est estimée par l'abscisse du point correspondant au centre de l'intervalle M pour lequel la fonction d'appartenance est maximale. Cette valeur est fournie par l'expression

$$u_r = (\inf(M) + \sup(M))/2 \quad (II.26)$$

Où $\inf(M)$ et $\sup(M)$ sont respectivement les bornes inférieure et supérieure de l'intervalle M .

II.3.6 Les Différents types de contrôleurs flous

II.3.6.1 Le contrôleur flou e type Mamdani

Un système flou de Mamdani est basé sur une collection de règles du type :

$$\text{Si } x \text{ est } A_i \text{ et } y \text{ est } B_j \text{ Alors } Z \text{ est } C_k$$

Où A_i , B_j , C_k représentent des termes linguistiques auxquels est attribuée une signification floue que l'on suppose normalisée pour des entrées numériques précises, x_0 et y_0 , un sous-ensemble flou de sortie, noté F , est généré selon la relation

$$\mu_F(z) = \max_{(i,j,k) \in I} \min (\mu_{A_i}(x_0), \mu_{B_j}(y_0), \mu_{C_k}(z)) \quad (II.27)$$

L'opération $\max_{(i,j,k) \in I}$ correspond à l'application de l'opérateur max sur l'ensemble des règles définies. La valeur numérique de sortie délivrée par le système flou est obtenue par défuzzification, classiquement implantée par la méthode du centre de gravité

$$z = \frac{\int z \cdot \mu_F(z) dz}{\int \mu_F(z) dz} \quad (II.28)$$

II.3.6.2 Le contrôleur flou e type Takagi-Sugeno

Il est construit à partir d'une base de règles "Si ... alors ...". Les prémisses sont toujours exprimées linguistiquement mais les conclusions sont par contre de nature numérique. Elles peuvent prendre différentes formes : équations linéaires ou polynômiales, statiques ou dynamiques.

L'étude est ici restreinte à des conclusions constantes, les règles étant donc formulées de manière suivante :

$$R_{(i,j,k)} : \text{Si } x \text{ est } A_i \text{ et } y \text{ est } B_j \text{ alors } Z = C_k$$

Où C_k représente une valeur réelle quelconque. Pour les valeurs d'entrée précise x_0 et y_0 , la sortie z est évaluée selon le mécanisme de calcul suivant :

$$Z = \text{SUG}(x_0, y_0) = \frac{\sum_{(i,j,k) \in I} w_{i,j} \cdot C_k}{\sum_{(i,j,k) \in I} w_{i,j}} \quad \text{avec} \quad w_{i,j} = u_{A_i}(x_0) \cdot u_{B_j}(y_0)$$

II.4 Avantages et Inconvénients de la logique floue

Parmi ses avantages, on cite :

- Facilité d'implantation.
- Solution de problèmes multi variables complexes.
- Robustesse vis à vis des incertitudes.
- Possibilité d'intégration du savoir de l'expert.
- Possibilités de commande auto-adaptative aux variations du procédé

Par contre elle a aussi des inconvénients, les plus rencontrés sont :

- Réglage empirique.
- Performances en boucle fermée dépendent de l'expertise.
- Il n'existe pas de théorie générale qui caractérise rigoureusement la stabilité, la robustesse.

II.5 Domaines d'applications

Parmi les nombreuses applications de la logique floue, la commande floue s'avère être le champ d'application le plus actif à travers le monde. Nous présentons quelques exemples d'applications de la commande floue [20]:

- Commande de tubes broyeurs pour la fabrication du ciment (première réalisation en 1979 au Danemark).
- Commande de métros avec un fonctionnement plus confortable et économique et une précision d'arrêt augmentée (1987 à Sendai, Japon).
- Production du fer et de l'acier, purification de l'eau, chaînes et robots de fabrication, opérateurs, unités arithmétique, micro-ordinateurs...etc.)
- Gestion des aspirateurs, des systèmes de ventilation et de régulation thermique.
- Lecture automatique, reconnaissance de caractères.
- Traitements d'images.
- Systèmes automobiles embarqués (ABS, suspension, climatisation, ...etc.)
- Systèmes de transport (train, métro, ascenseur, ... etc),
- Systèmes de décision, diagnostic (médical, assurance).

II.6 Conclusion

Dans ce chapitre, nous avons présentés les notions de base de la logique floue ainsi que la composition d'un système de commande floue. Les notions fondamentales du raisonnement flou ont été éclaircies en expliquant comment s'effectue la déduction des nouvelles informations à partir des informations déjà disponibles à base des règles linguistiques.

Le concept d'ensembles flous peut être appliqué à plusieurs problèmes, selon la nature de l'information, la manipulation de l'imprécis et la complexité de problèmes.

CHAPITRE III
SIMULATION & INTERPRETATION

Chapitre III : Simulation & Interprétations

III.1 Introduction

Le rôle de l'automaticien (chargé d'obtenir un système régulé) sera multiple :

- Instrumenter le système, c.-à-d., choisir les capteurs et actionneurs en fonction des besoins physiques, de coût et de performances demandées au système.
- Déterminer les relations entrées-sorties du système, des capteurs et des actionneurs.

On parlera dès lors de :

- ☞ **modéliser** quand on s'attachera à déterminer la structure mathématique de ces relations.
- ☞ **identifier** quand on s'intéressera à calculer les coefficients du modèle.
- ☞ **Synthétiser** une loi de commande (un correcteur) afin d'obtenir un système performant : Précis, rapide et stable, tout en s'affranchissant des influences néfastes des perturbations.

Le système ainsi corrigé (asservi, régulé) devra assurer deux objectifs :

- ❖ **la poursuite** : suivre une entrée de consigne (référence). On désire asservir la sortie à l'entrée (la sortie doit « ressembler » le plus possible à l'entrée) et ainsi assurer des performances (stabilité, rapidité, précision).
- ❖ **La régulation** : annuler (ou diminuer) les effets de la (ou des) perturbation(s).

Dans ce chapitre, on va implémenter deux types de commande : une commande PID et une autre commande dite floue pour différents systèmes linéaires représentés par des fonctions de transfert de 1^{er} ordre, du 2^{ème} ordre et de 3^{ème} ordre afin de comparer les résultats et les performances de chacune. Pour mettre en œuvre ces commandes, on a utilisé le SIMULINK de MATLAB dont le manuel de son utilisation dans le cas de la commande floue est décrit dans ANNEX.

III.2 Application de La Commande PID

III.2.1 Système 1^{er} ordre : Régulation de la température d'une serre

III.2.1.1 Présentation du système

On s'intéresse dans ce système du réglage de la température d'une serre. Pour cela, la température est mesurée avec un capteur qui fournit la valeur de la grandeur à régler. On suppose que le système est linéaire et ayant une fonction de transfert $H(s) = 1/(s+2)$ avec un retard égale à 1s. C'est un système de contrôle de température avec un élément de chauffage et de ventilation. La température désirée est de 20°C. La température extérieure est de 10°C et son effet intervient au niveau de l'intégrateur. On modélise le système avec SIMULINK de MATLAB.

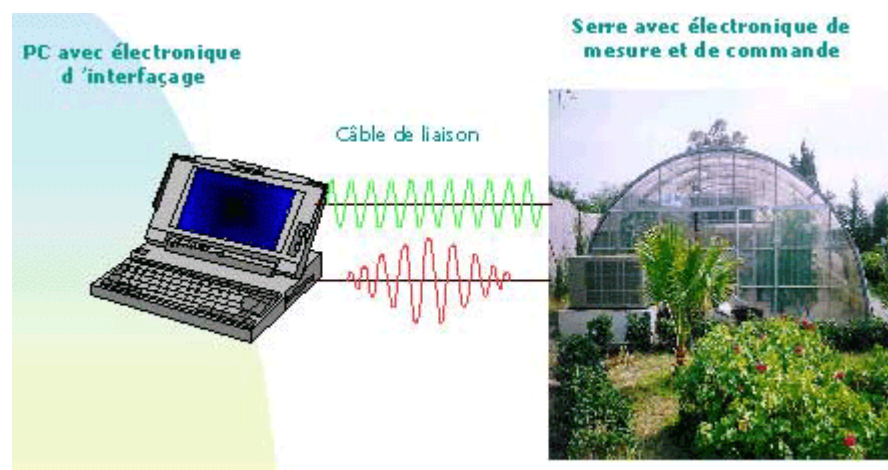


Fig. III.1 – Méthodologie de commande

III.2.1.2 Commande du système

Le système est commandé en boucle fermée. Les paramètres du système simulé sont : Step (référence)=20°C ; Step1(perturbation)=1.5°C ; Transport Delay (retard)=1 ; PID (Kp=1 ;

$K_i=0$; $K_d=0$). Les paramètres du PID sont obtenu par essais successifs jusqu'à obtention de bons résultats.

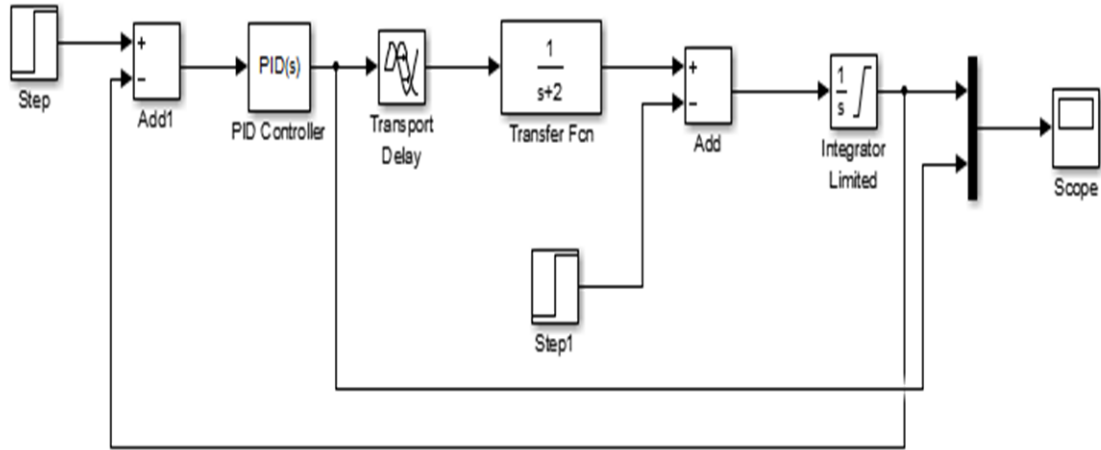


Fig. III.2 – Schéma bloc de la boucle de régulation PID de la température.

Après simulation du schéma fonctionnel de la figure III.2, et on a trouvé les résultats suivants :

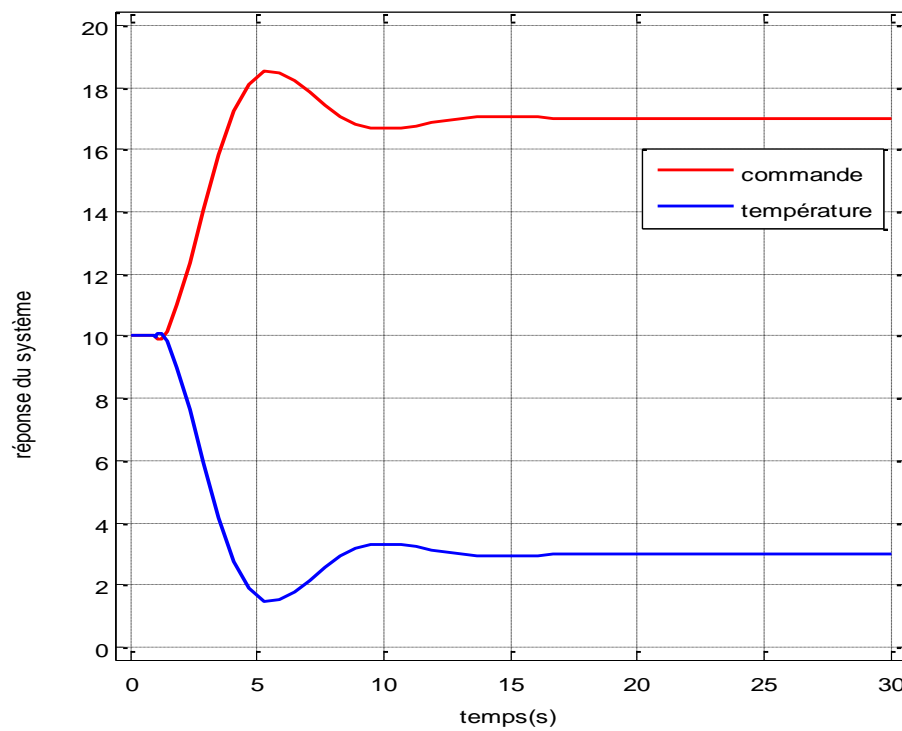


Fig. III.3 – La réponse du système de 1^{er} ordre commandé par PID

La réponse du système présente une erreur importante, malgré que le temps de réponse est court ainsi que le système est stable dans le régime permanent. Pour remédier à ces problèmes, on se propose de réguler le système avec un régulateur flou.

III.2.2 Système de 2ème ordre : Régulation d'un moteur à courant continu

III.2.2.1 Présentation du système

Selon le schéma de la Figure(III.4), un moteur électrique à courant continu est régi par les équations physiques découlant de ses caractéristiques électriques, mécaniques et magnétiques. D'après la loi de Newton, combiné à des lois de Kirchhoff, On peut écrire les équations différentielles de premiers ordres après a transformation de Laplace comme suit :

$$p(J.p + b)\theta(p) = K.I(p) \quad (\text{III.1})$$

$$(L.p + R)I(p) = V - K.p.\theta(p) \quad (\text{III.2})$$

$$F(p) = \frac{\theta}{V} = \frac{K}{(J.p+b)(L.p+R)+K^2} \quad (\text{III.3})$$

Avec :

- R : résistance électrique = 1 ohm
- L : Inductance électrique = 0.5 μH
- K : constante de force électromotrice = 0.01 Nm/Amp
- J : moment d'inertie du rotor = 0.01 $\text{kg m}^2/\text{s}^2$
- b : Rapport d'humidité = 0.1 Nms
- Entrée : la source de tension (V)
- Sortie : la vitesse de rotation (θ)

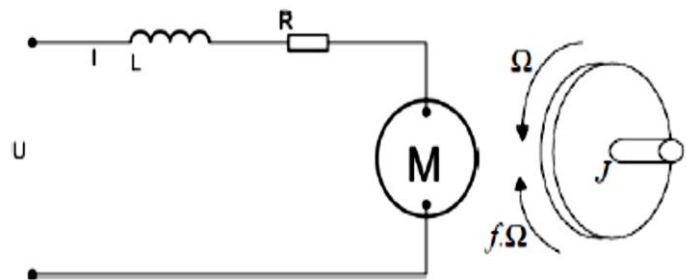


Fig. III.4- Schéma équivalent d'un moteur à courant continu.

Après le calcul, le moteur à courant continu est modélisé par la fonction du transfert suivante :

$$F(P) = \frac{0.01}{0.005s^2 + 0.06s + 1.001} \quad (\text{III.4})$$

III.2.2.2 Commande du système

- La première étape consiste à étudier le système en boucle fermée sans et avec le contrôleur PID pour voir son comportement avec et sans contrôleur. Les paramètres du système simulé sont : Step (consigne)=10 rad/s ; Paramètres du PID : $K_p = 200$, $K_I = 50$, $K_D = 1$

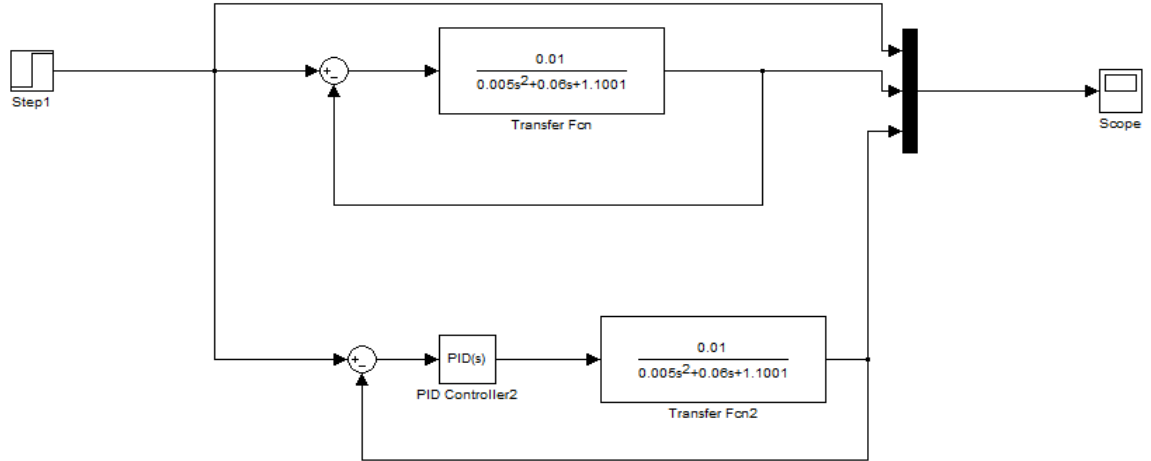


Fig. III.5 – Schéma bloc du système de 2^{ème} ordre en boucle fermée avec et sans PID.

On a réalisé le schéma de la figure III.5 sous Matlab- SIMULINK et on a trouvé les résultats suivants :

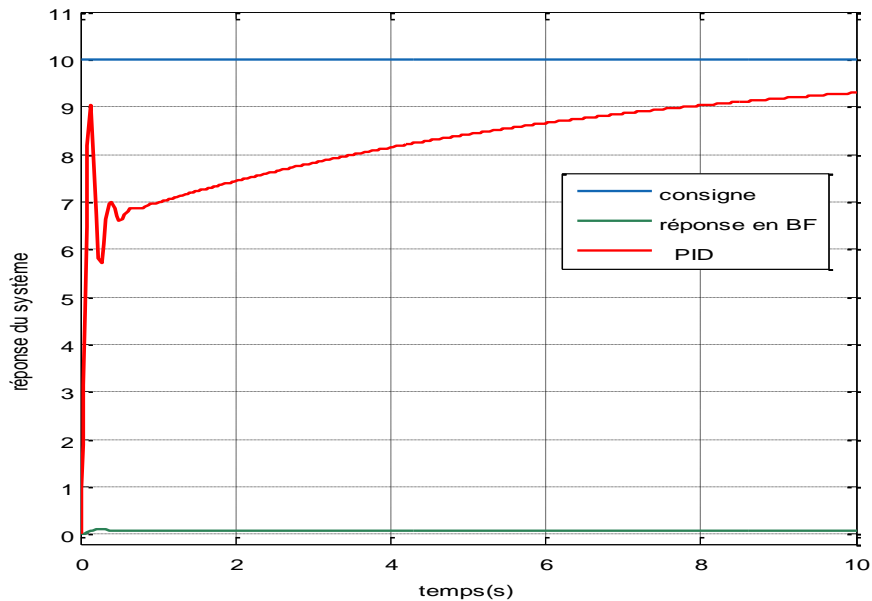


Fig. III.6 – Réponses du système de 2^{ème} ordre avec et sans PID

Le système en boucle fermée présente une forte erreur en régime permanent tandis que le système commandé par le PID s'améliore dans le côté précision mais n'atteint pas la référence voulue.

- ☞ **La deuxième étape** consiste à étudier le système commandé par le PID avec présence de perturbation. Les paramètres du système simulé sont : Step2 (référence)=10 ; step4 (perturbation)=1.5 ; Paramètres du PID : $K_p = 200$, $K_I = 50$, $K_D = 1$

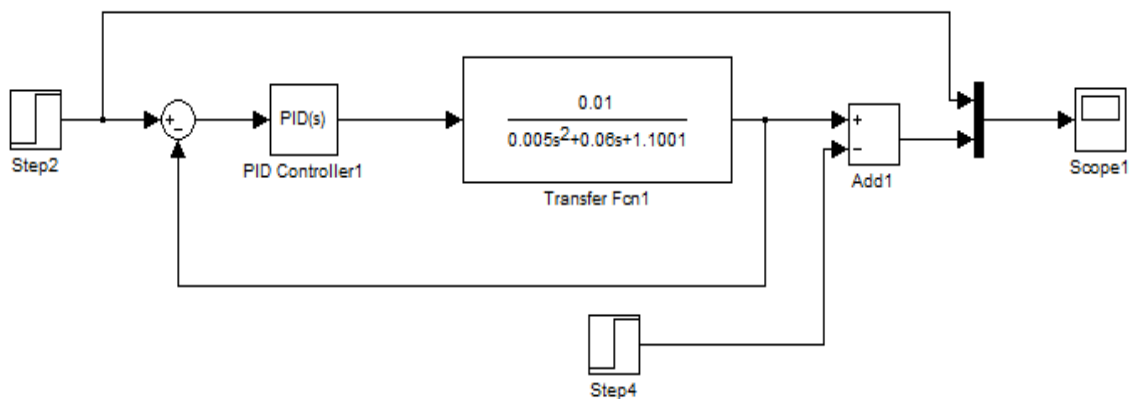


Fig. III.7 – Réponse du système de 2^{ème} ordre commandé par PID avec perturbation.

On a réalisé le schéma de la figure sous Matlab- SIMULINK et on a trouvé les résultats suivants :

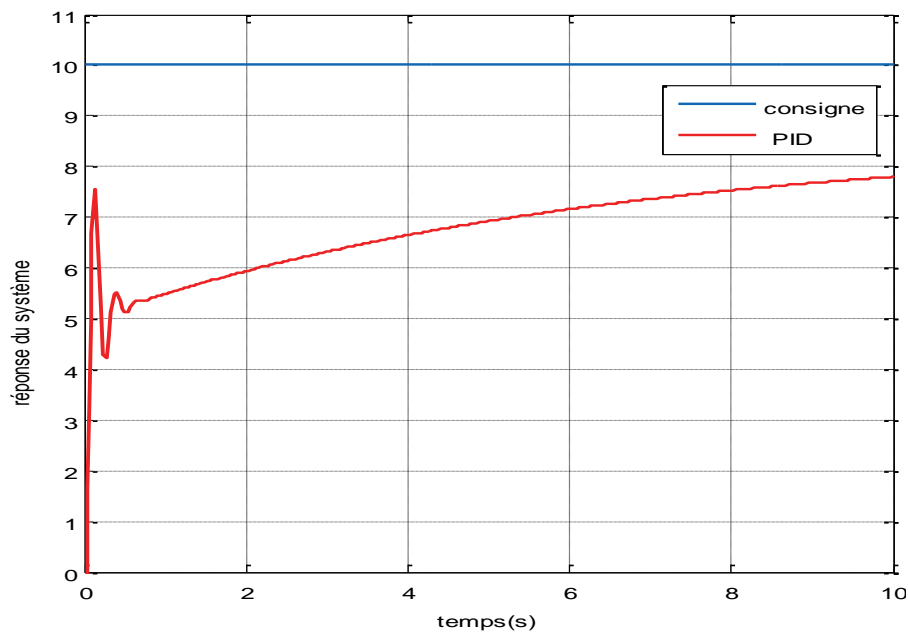


Fig. III.8 – Réponse du système de 2^{ème} ordre avec perturbation commandé par PID

La réponse du système avec perturbation s'éloigne encore plus que celle sans perturbation, ainsi que le temps de montée est très grand.

III.2.3 Système de 3^{ème} ordre :

III.2.3.1 Présentation et commande du système

Le système concerné est caractérisé par la fonction de transfert suivante :

$$F(p) = \frac{3}{s^3 + 6s^2 + 11s + 6} \quad (III.5)$$

Le système est étudié en boucle fermée avec et sans PID. Les paramètres du système simulé sont : Step1=10 ; Paramètres du PID : $K_p = 5$, $K_I = 3$, $K_D = 2$

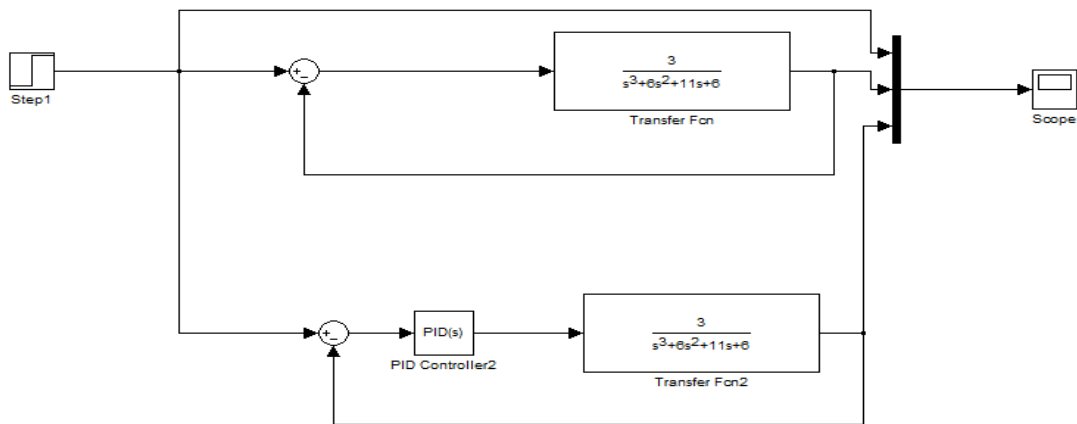


Fig. III.9 – Schéma bloc d'un système de 3^{ème} ordre en boucle fermée avec et sans PID.

Le schéma de la figure III.9 est réalisé sous Matlab- SIMULINK. Les résultats trouvés sont les suivants :

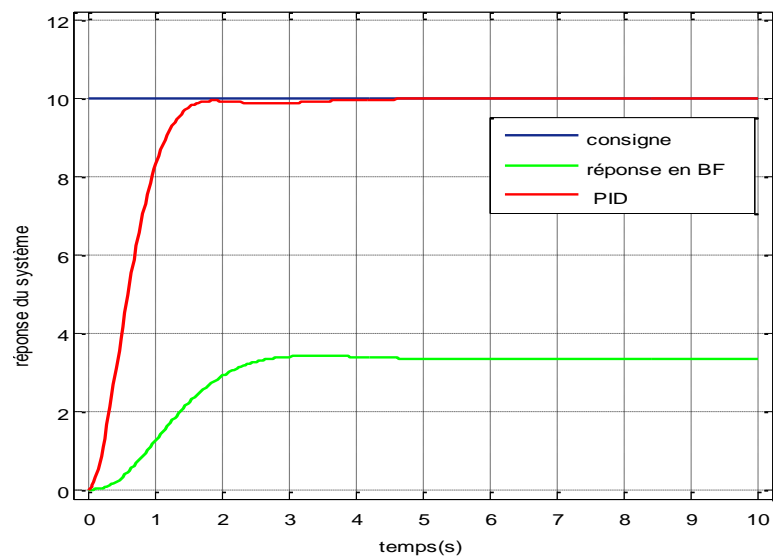


Fig. III.10 – La réponse du système de 3^{ème} ordre commandé par PID.

La réponse du système en boucle fermée sans contrôleur présente une forte erreur en régime permanent, mais avec le PID elle présente une bonne poursuite de la consigne (erreur nulle).

- La deuxième étape consiste à étudier le système commandé par le PID avec la présence de perturbation. Les paramètres du système simulé sont : $\text{Step2}=10$; $\text{step4}=3$, Paramètres du PID : $K_p = 17$, $K_I = 0$, $K_D = 7$.

On a réalisé le schéma de la figure sous Matlab- SIMILINK et on a trouvé les résultats suivants :

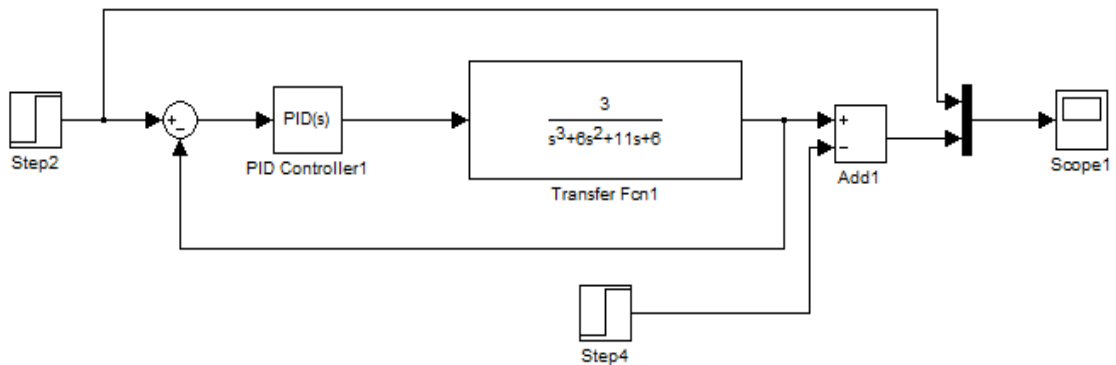


Fig. III.11 – Schéma bloc d'un système 3eme ordre en boucle fermée et avec un PID

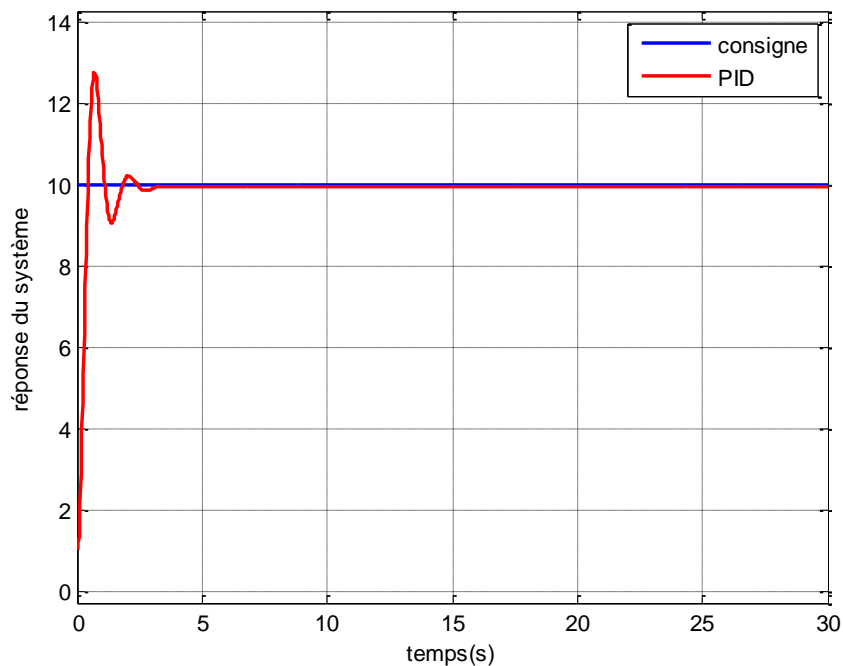


Fig. III.12 – Réponse du système de 3eme ordre avec perturbation commandé par PID

La réponse du système commandé par le PID présente un dépassement de 25% bien que l'erreur en régime permanent soit nulle.

III.3 Application de La Commande Floue

III.3.1 Etapes pour créer le régulateur flou en utilisant la boîte à outil fuzzy-logic :

fuzzy-logic :

Pour configurer le contrôleur flou dans le Simulink de MATLAB, il faut passer par les étapes suivantes :

- On commence par ouvrir *Matab Command*, et taper `>>fuzzy`, la boîte à outil de « fuzzy logic » va s'ouvrir. Après on suit les étapes suivantes :
- **Edit Add input**

On commence tout d'abord par définir les entrées et les sorties, et par la suite on va essayer de lier les fonctions d'appartenances d'une façon logique. On a choisi le système de 1^{er} ordre pour mettre en œuvre toutes les étapes pour construire le contrôleur flou.

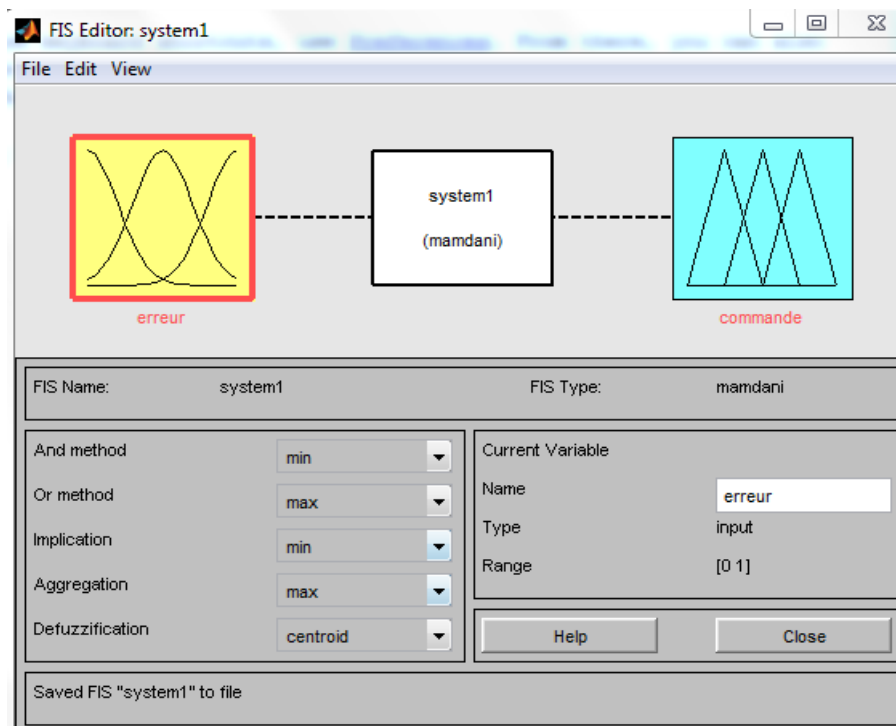


Fig. III.13 – Création des fonctions d'appartenance des entrées et de sortie.

- **View Edit Membership function**
- **Edit Add Mfs** (pour l'entrée :erreur)

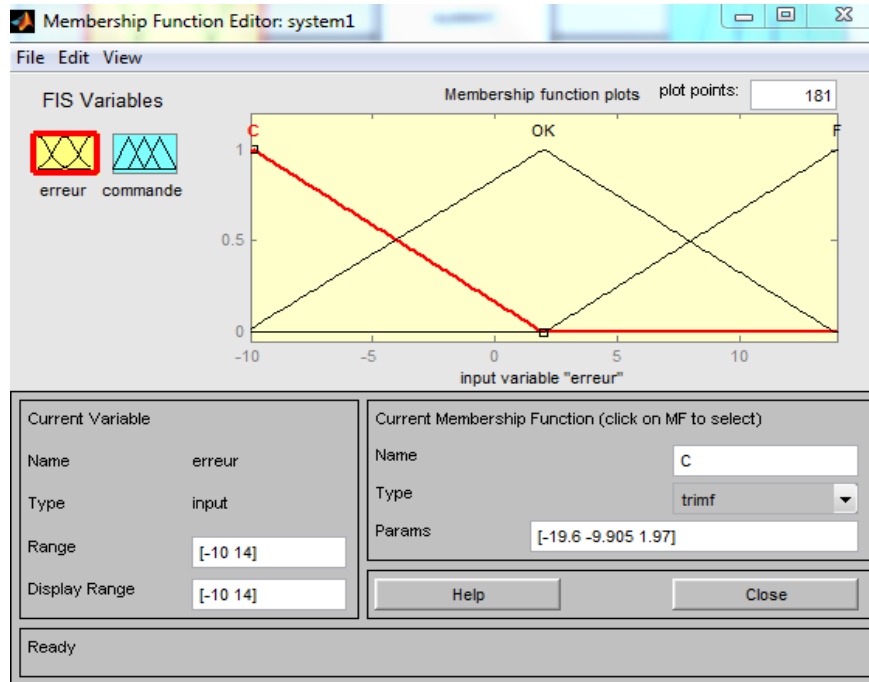


Fig. III.14 – Fuzzification de l'entrée du contrôleur flou

- **View Edit Membership function**
- **Edit Add Mfs** (pour la sortie :commande)

La commande a été partitionnée en 3 valeurs linguistiques triangulaires et sémitriques, Elle peut être sur un intervalle de [-30,37] comme pour le système de 1^{er} ordre.

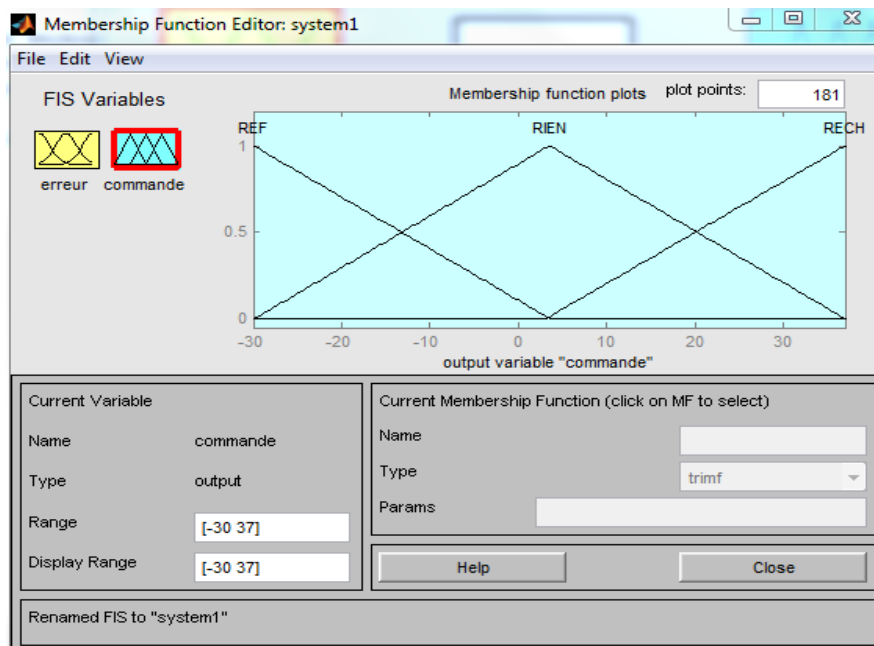


Fig. III.15 – Fuzzification de la sortie du contrôleur flou

➤ **View Edit rule**

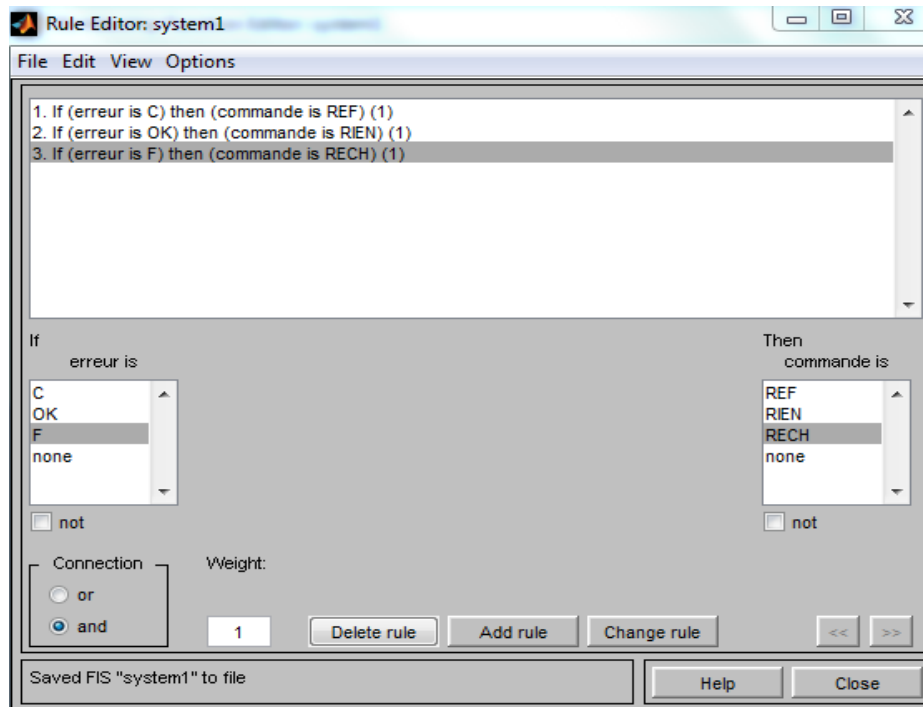


Fig. III.16 – les règles floues

➤ **View view rule**

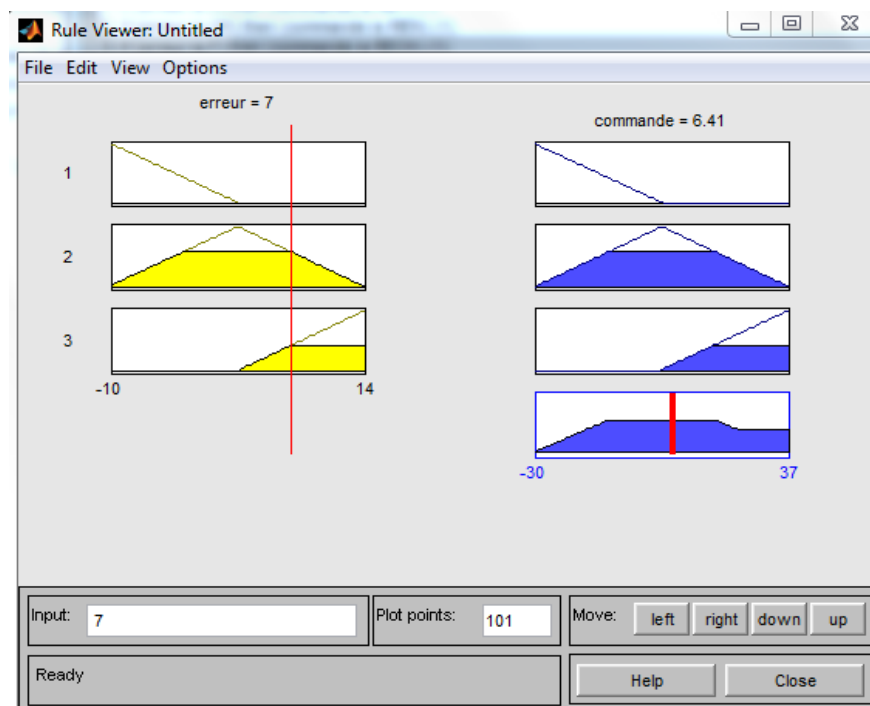


Fig. III.17 – l'agrégation des règles

➤ View view surface

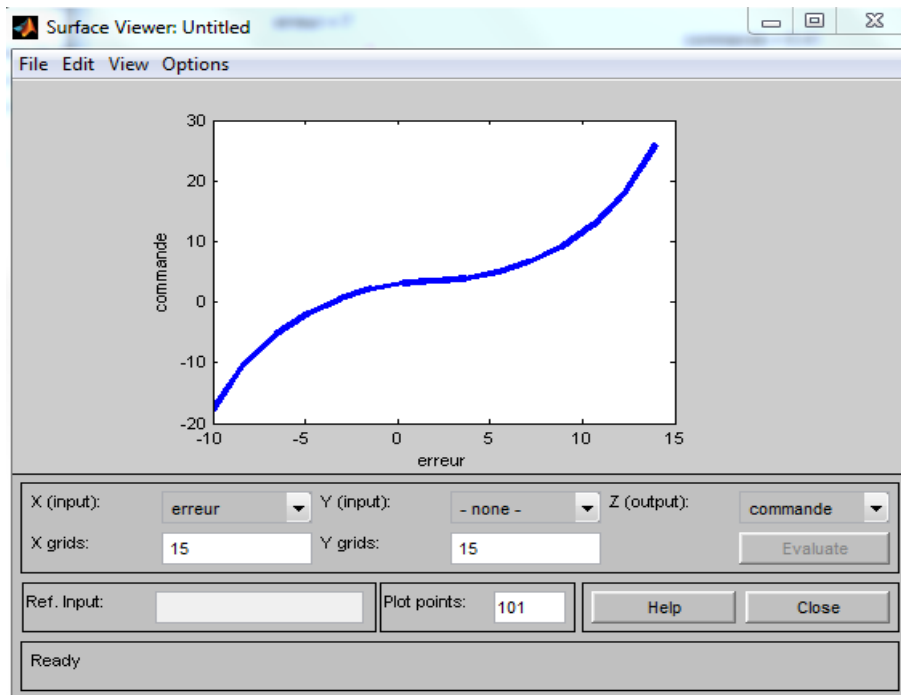


Fig. III.18 – Surface des règles floues

III.3.2 Système de 1^{er} ordre : Régulation de la température d’une serre

III.3.2.1 Régulateur flou à une seule entrée

On remplace le contrôleur PID par un contrôleur flou à une seule entrée :

- L’entrée du contrôleur est l’erreur entre la sortie désirée et sortie calculée.
- La sortie du contrôleur est la commande du système.

On considère alors le système avec perturbation présenté dans la figure III.19.

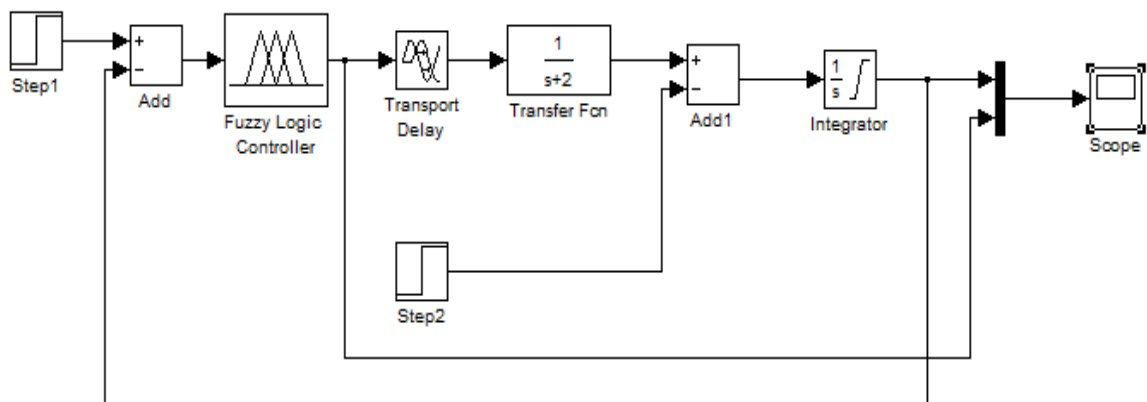


Fig. III.19 – Schéma bloc de la boucle de régulation à contrôleur flou à une seule entrée.

- **La fuzzification**

Pour construire notre régulateur flou, on commence tout d'abord par définir les entrées et les sorties, On prendra comme entrée l'erreur, c'est l'écart de la température mesurée et désirée, et pour sortie la commande (en utilisant l'interface fis Editor). Ensuite on définira les plages de variations (les ensembles flous) et les fonctions d'appartenances en utilisant l'interface (Edit Membership function) ; et ceci et pour l'entrée et pour la sortie. Les plages de variation des entrées et de la sortie sur le régulateur sont :[-10 14] pour l'erreur et [-30 37] pour la commande.

Les fonctions d'appartenances utilisées sont de type trimf (triangulaires).

- **Les inférences**

Une fois le type de fonctions d'appartenance choisi ainsi que la plage de variation, on appellera chaque partie de la fonction d'appartenance par un nom significatif (label linguistique), on aura alors :

Lorsque l'erreur est négative, la température de sortie est supérieure à la consigne donc c'est **chaud**, Lorsqu'elle est positive la fonction d'appartenance est nommée **froid**, et la partie du milieu sera nommée **OK**.

On fera de même pour la commande, la partie décroissante sera nommée **refroidir** (Elle correspond à l'entrée chaud), la partie croissante sera nommée **réchauffer**, et les deux segments du milieux sont assignés à une commande appelée **rien**.

Les règles d'inférences seront alors définies comme suit :

R1 : Si c'est chaud alors refroidir.

R2 : Si c'est Ok alors rien faire.

R3 : Si c'est froid alors réchauffer.

Le système d'inférences floues sera de type Mamdani et la **défuzzification** se fera avec la méthode de centre de gravité (centroïde).

On enregistre le régulateur sur le disque et sur le *workspace* pour pouvoir le charger à partir de Simulink.

Ceci nous a permis d'avoir la réponse suivante :

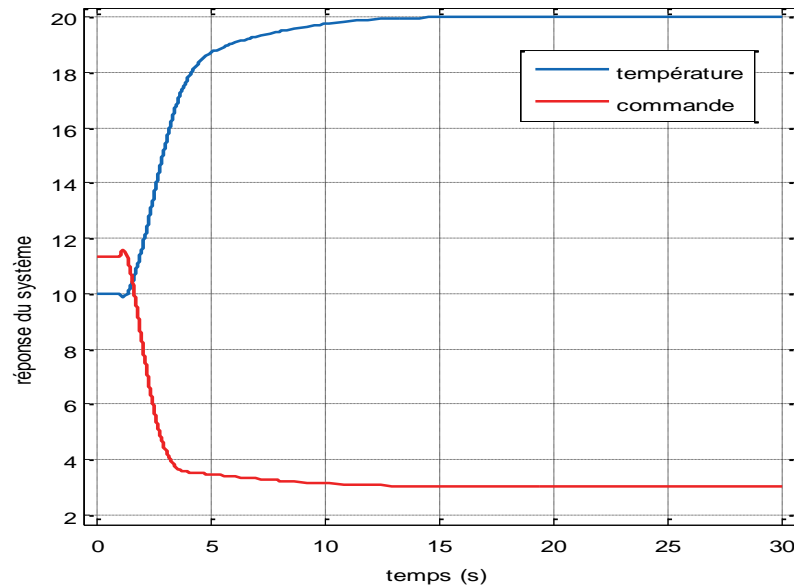


Fig. III.20 – Réponse du système du 1^{er} ordre commandé par un régulateur flou à une seule entrée

C'est une réponse qui atteint la consigne (20 °C) et y reste. En effet en variant les plages d'inférences on a augmenté l'efficacité du régulateur autour de cette consigne. On pourrait également modifier les règles d'inférences ou les formes des fonctions d'appartenances utilisées, mais ceci exige une maîtrise du système à réguler.

III.3.2.2 Régulateur flou à deux entrées

On se propose dans cette partie d'ajouter une entrée au régulateur flou, c'est la variation de l'erreur en fonction du temps, ou la dérivée de l'erreur.

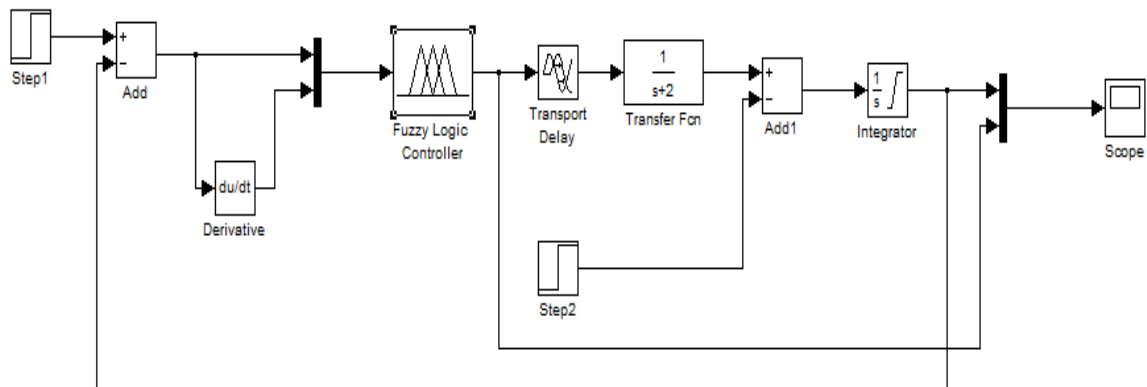


Fig. III.21 – Schéma bloc de la boucle de régulation à contrôleur flou à 2 entrées

Une fois le régulateur est configuré, on simule et on obtient la réponse suivante :

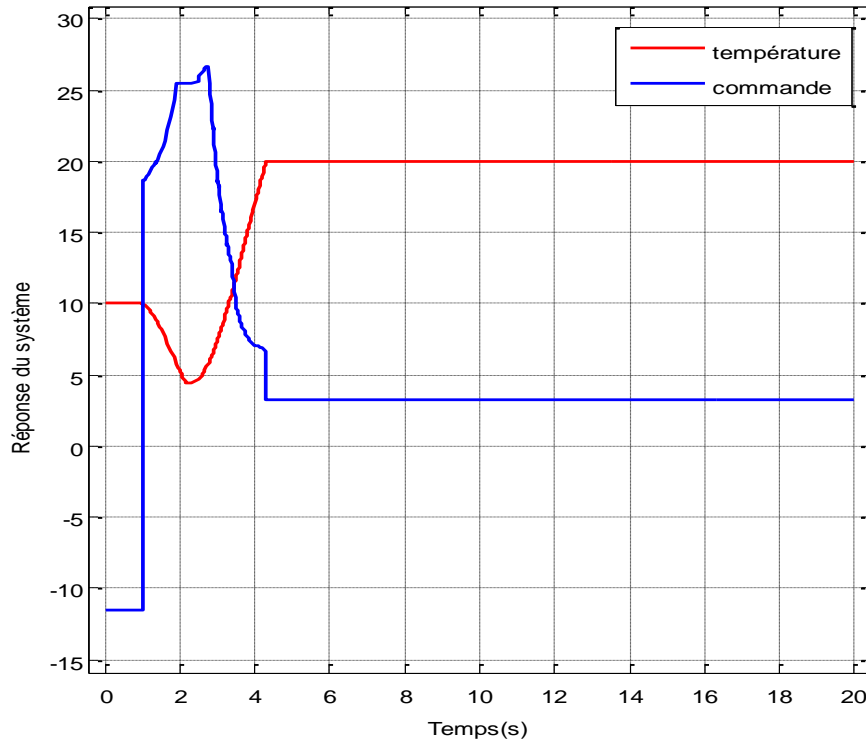


Fig. III.22 – Réponse du système du 1^{er} ordre commandé par un régulateur flou à 2 entrées

Cette réponse est obtenue également après modification des plages de variations (univers de discours) : [-10 14] pour l'erreur, [-15 15] pour la dérivé de l'erreur et [-30 44] pour la commande. On a utilisé des fonctions d'appartenances de type triangulaires pour les 2 entrées ainsi que la sortie. La base de règles est composée de 9 règles (voir ANNEX).

La réponse présente certes un dépassement mais sa rapidité est meilleure : son temps de réponse est inférieur à celui du premier cas. Dans l'ANNEX nous présentons toutes les interfaces qui ont servi à construire ce régulateur.

III.3.3 Commande Floue du système de 2^{ème} ordre

On remplace le contrôleur PID de la figure III.5 par un contrôleur flou à 2 entrées :

- Les entrées du contrôleur sont l'erreur (écart entre la sortie désirée et sortie calculée), et la dérivée de l'erreur.
- La sortie du contrôleur est la commande du système.

III.3.3.1 Commande du système en poursuite

On considère alors le système avec contrôleur flou présenté dans la figure III.23.

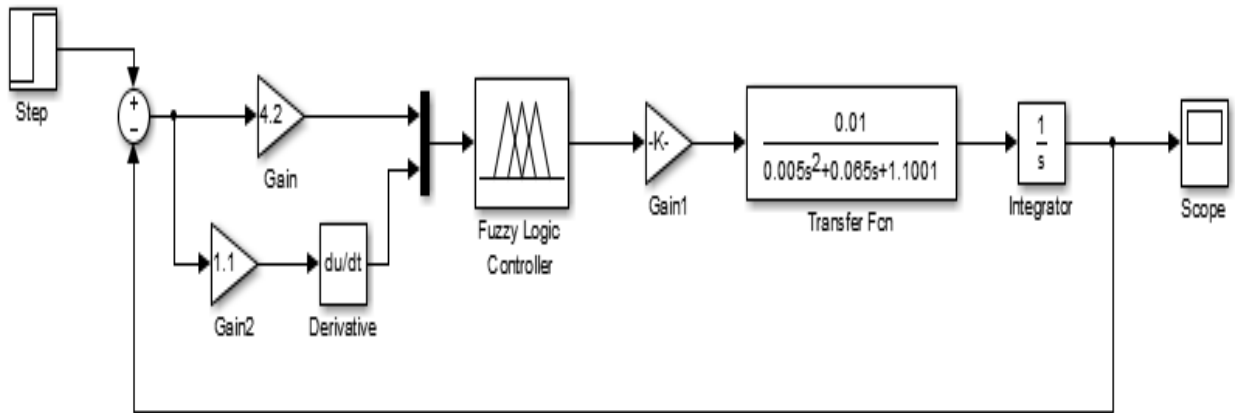


Fig. III.23 – Schéma bloc de système de 2eme ordre commandé par le contrôleur flou.

Les plages de variation des entrées et de la sortie pour le régulateur flou sont : [-20 20] pour l'erreur, [-10 10] pour a dérivée de l'erreur et [-75 75] pour la commande. Les fonctions d'appartenances des entrées et de sortie sont de type triangulaire à 3 ensembles flous. La base de règles est choisie comme celle de MacVicar-Whelan à 9 règles représentée par le tableau III.1.

u		e		
		N	Z	P
Δe	N	N	N	Z
	Z	N	Z	P
	P	Z	P	P

Tab. III.1- Table des règles floues du contrôleur flou (333)

Le système d'inférences floues sera de type Mamdani et la **défuzzification** se fera avec la méthode de centre de gravité.

Après simulation, on trouve les résultats suivants :

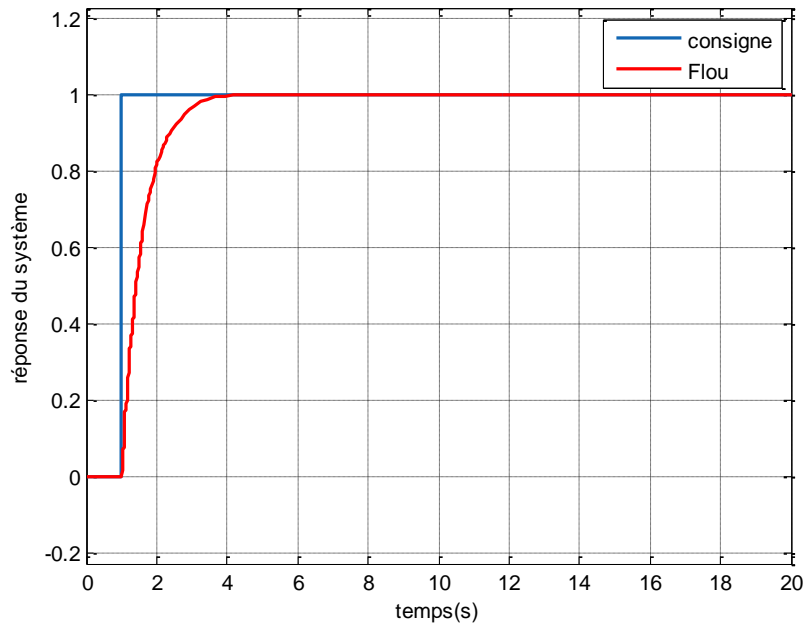


Fig. III.24 – La réponse du système de 2^{ème} ordre avec le contrôleur flou à 2 entrées

On remarque que la réponse du système avec le contrôleur flou atteint la consigne donc une bonne poursuite (erreur en régime permanent nulle), dans un temps de réponse très court et une marque la stabilité du système dans le régime permanent.

III.3.3.2 Commande en régulation

Après modification des plages de variations (univers de discours): [-30 30] pour l'erreur, [-10 10] pour la dérivé de l'erreur et [-100 100] pour la commande. On a utilisé des fonctions d'appartenances de type triangulaires pour les 2 entrées ainsi que la sortie. La base de règles est composée de 9 règles. La perturbation est de 10% du signal d'entrée.

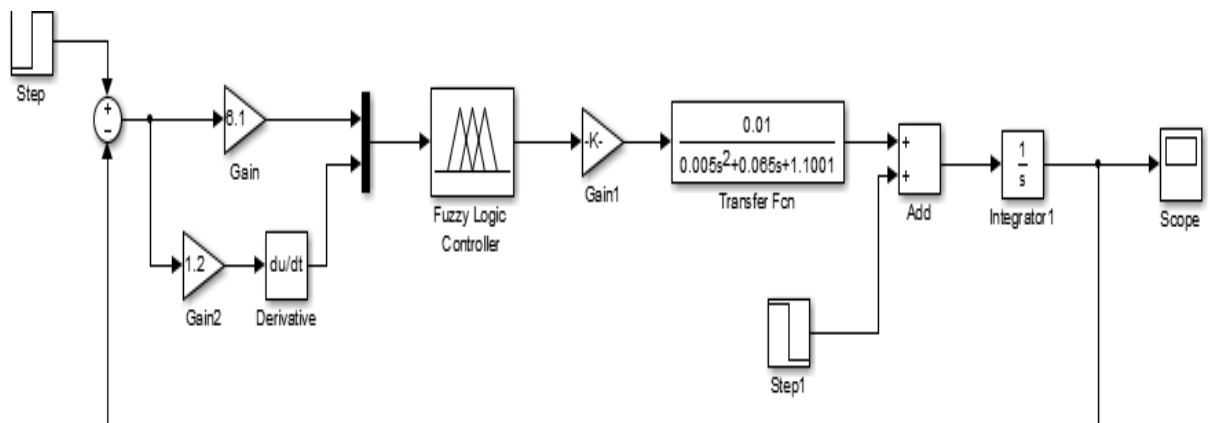


Fig. III.25 – Schéma bloc de commande en régulation par le contrôleur flou

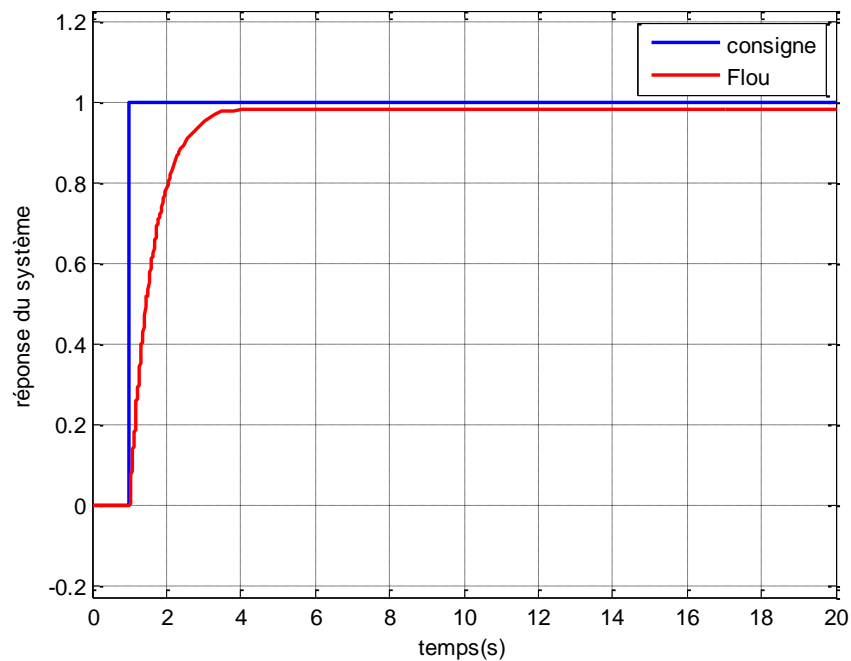


Fig. III.26 – La réponse du système de 2^{ème} ordre avec le contrôleur flou à 2 entrées avec perturbation.

Les résultats montrent une bonne convergence vers la référence du système (sortie désirée) malgré la présence de perturbation.

III.3.4 Régulation du système 3eme ordre à une seule entrée

III.3.4.1 Commande en poursuite du système

On remplace le contrôleur PID par un contrôleur flou à une seule entrée :

- L'entrée du contrôleur est l'erreur entre la sortie désirée et sortie calculée.
- La sortie du contrôleur est la commande du système.

On considère le système de 3^{ème} ordre commandé par le contrôleur flou, comme il est montré dans la figure III.27.

Les plages de variation de l'entrée et de la sortie pour le régulateur flou sont : [-10 10] pour l'erreur et [-23 23] pour la commande. Les fonctions d'appartenance de l'entrée et de la sortie sont de type triangulaire à 3 ensembles flous. La base de règles est composée de 3 règles.

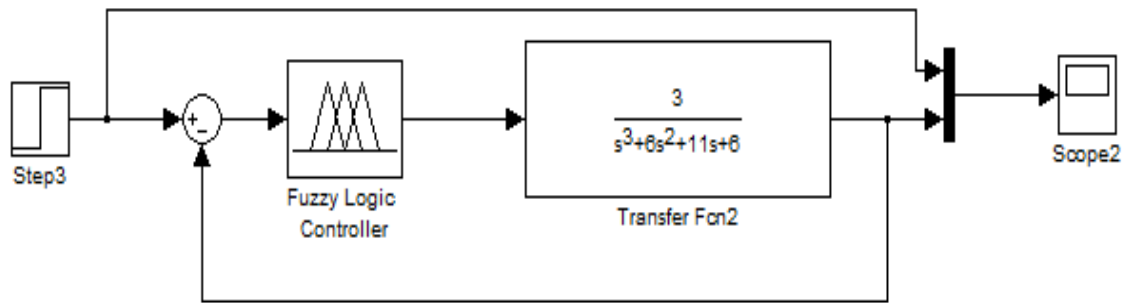


Fig. III.27 – Schéma bloc de système de 3^{ème} ordre commandé par le contrôleur flou

Après simulation, on trouve es résultats suivants :

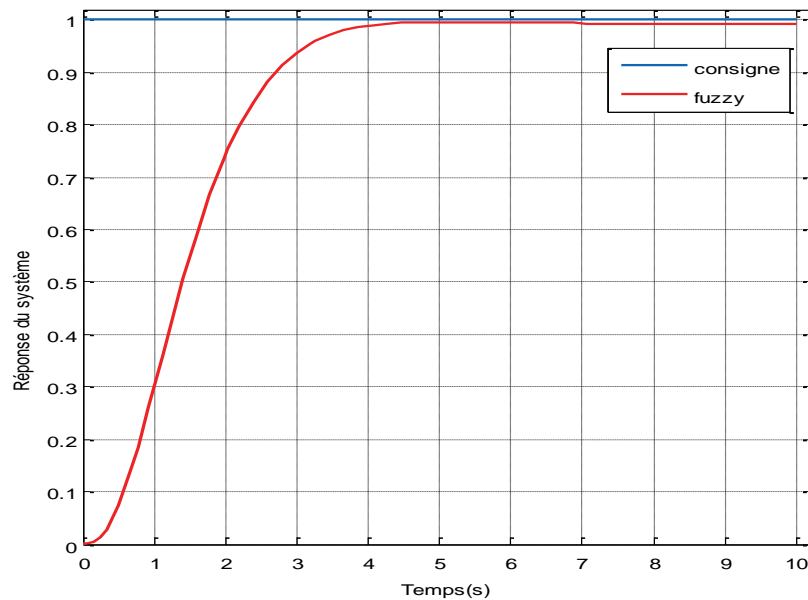


Fig. III.28– La réponse du système de 3^{ème} ordre soumis au contrôleur flou

Les résultats montrent une bonne convergence de la sortie vers le signal de consigne avec une bonne précision, rapidité et stabilité.

III.3.4.2 Régulation du système (avec perturbation)

On a gardé les mêmes plages de variation de l'erreur et de la commande pour le régulateur flou : [-10 10] pour l'erreur et [-23 23] pour la commande. Le schéma bloc considéré est celui de la figure III.29. La perturbation est de 5% du signal d'entrée.

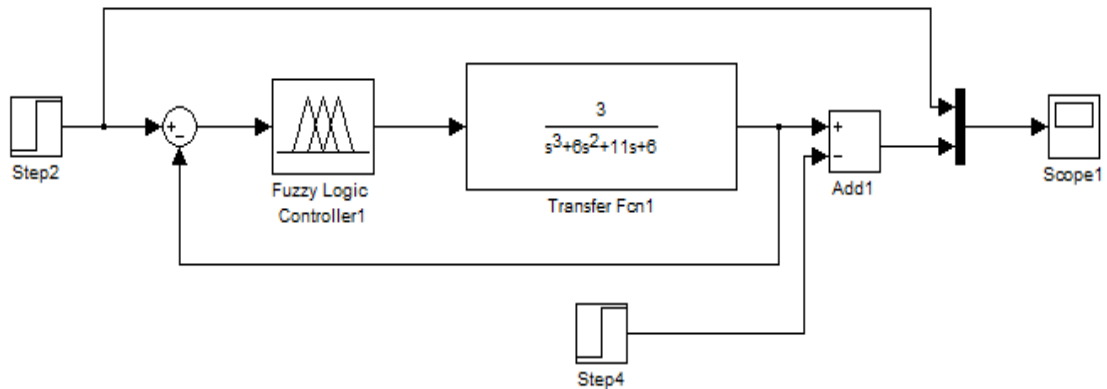


Fig. III.29– Schéma bloc de système de 3^{ème} ordre commandé en régulation.

Après simulation, on trouvé les résultats suivants :

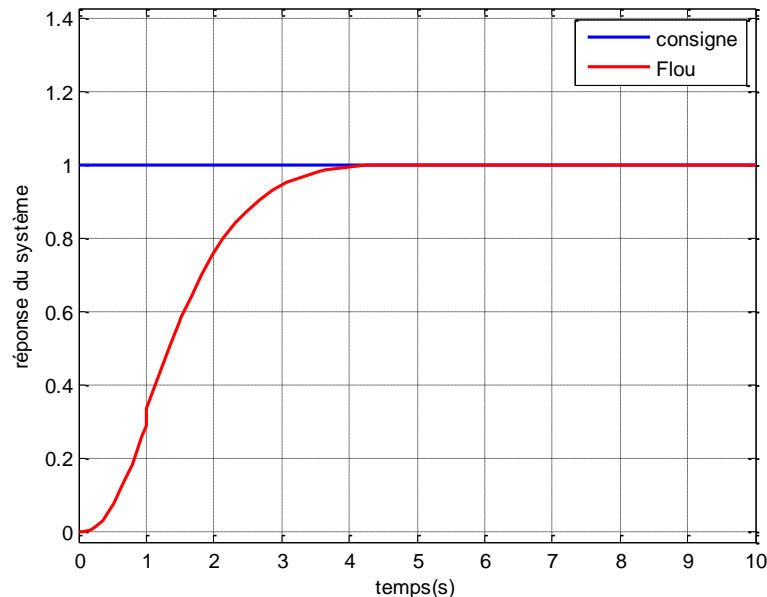


Fig. III.30 – La réponse du système de 3^{ème} ordre soumis au contrôleur flou avec perturbation.

- Les résultats montrent une bonne convergence vers le signal de référence (sortie désirée) malgré la présence de perturbation.

III.4 Comparaison entre La commande Floue et la Commande PID

Il est clair que l'avantage principal du contrôleur flou (CF) par rapport au contrôleur PID est le fait de pouvoir numériser le CF ou de le modéliser par ordinateur ou par

microprocesseur et par conséquent ce sera un régulateur évolutif que nous pouvons modifier à la demande, ce qui n'est pas le cas du PID. Pour procéder à la comparaison des résultats nous avons jumelé les schémas des simulations précédentes pour chaque système. Les résultats de la comparaison entre les régulateurs PID et CF sont représentés sur les figures III.32, III.34, III.36.

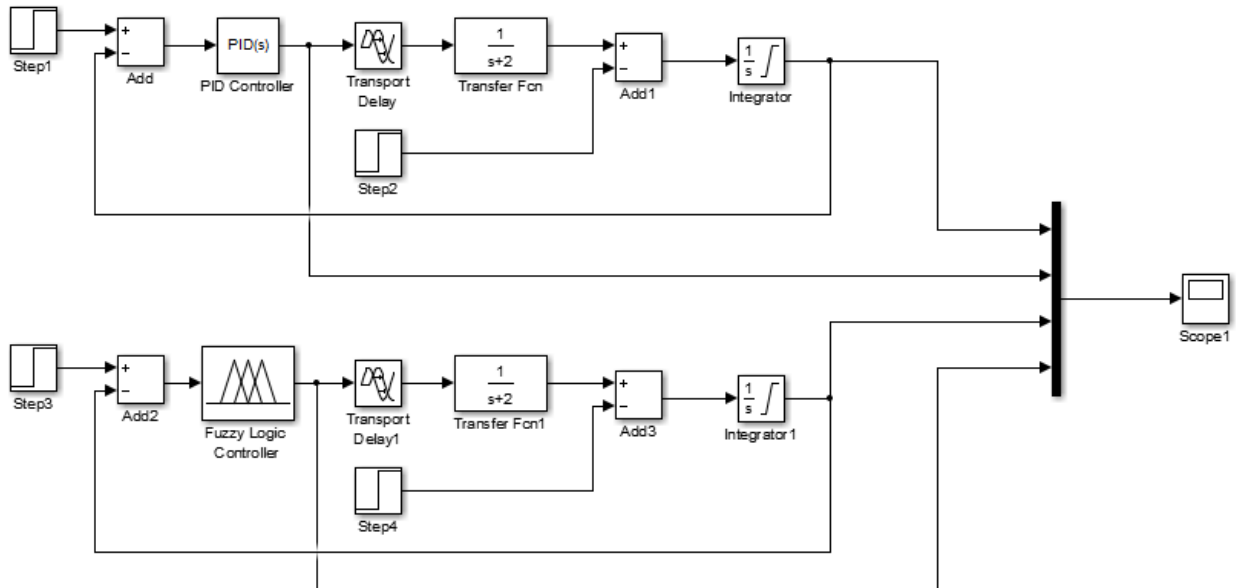


Fig. III.31 – Schéma bloc de système du 1^{er} ordre commandé par le PID et le contrôleur flou

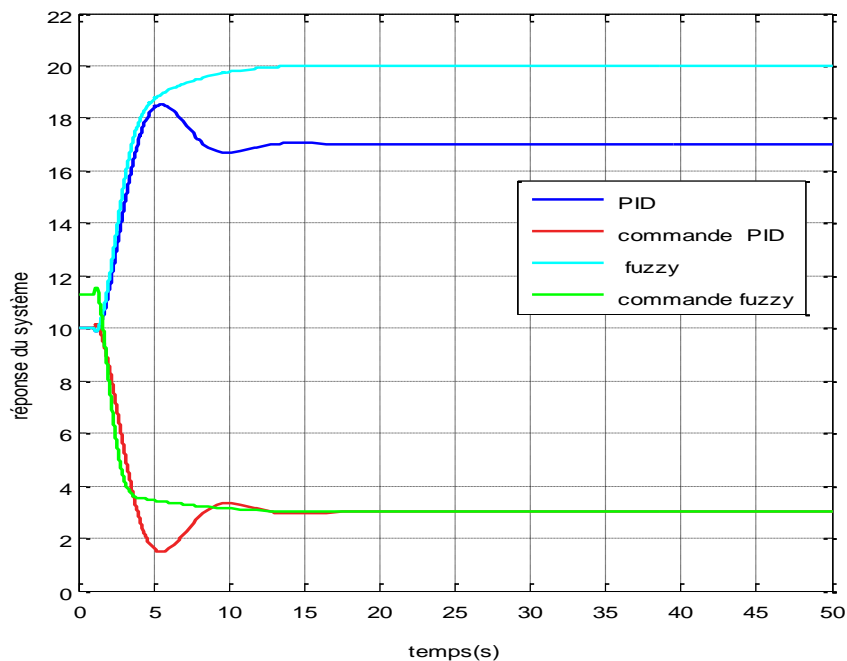


Fig. III.32 – La réponse du système du 1^{er} ordre.

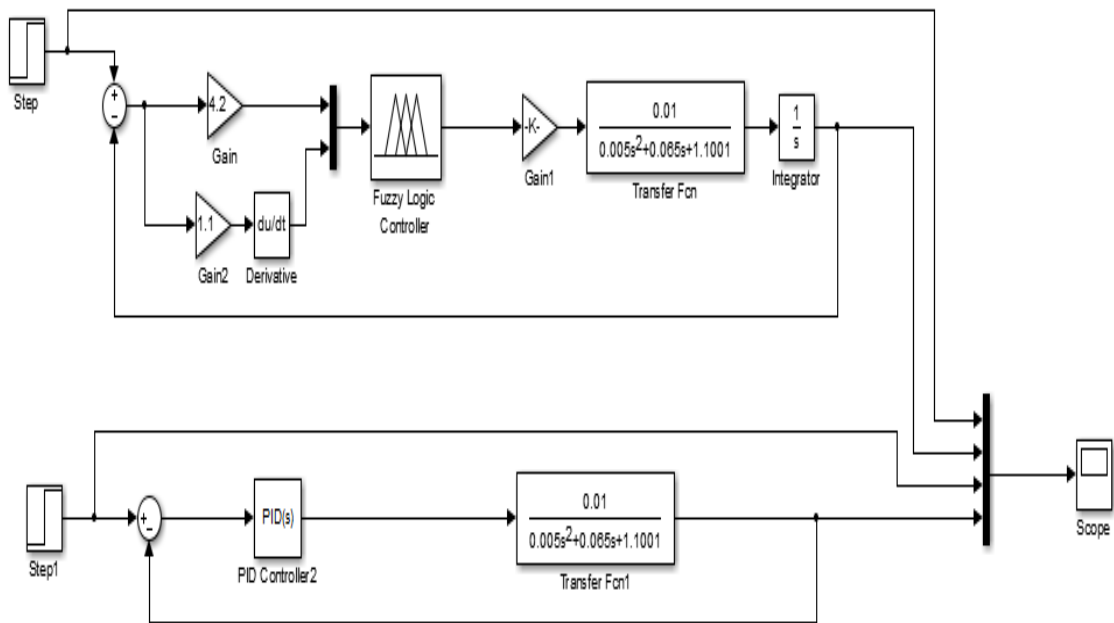


Fig. III.33 – Schéma bloc de système du 2^{ème} ordre commandé par le PID et le contrôleur flou

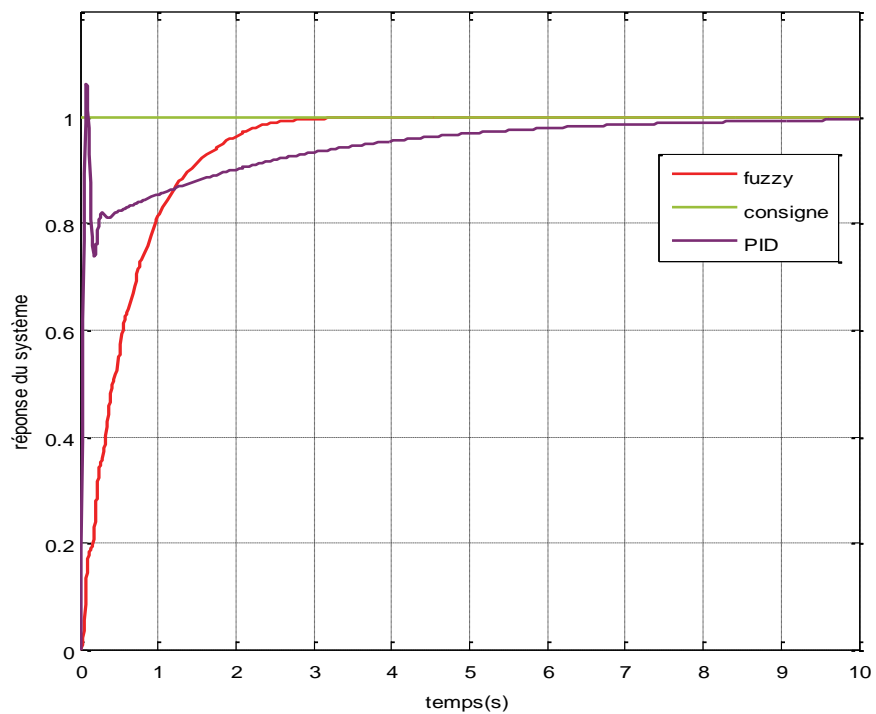


Fig. III. 34– La réponse du système du 2^{ème} ordre

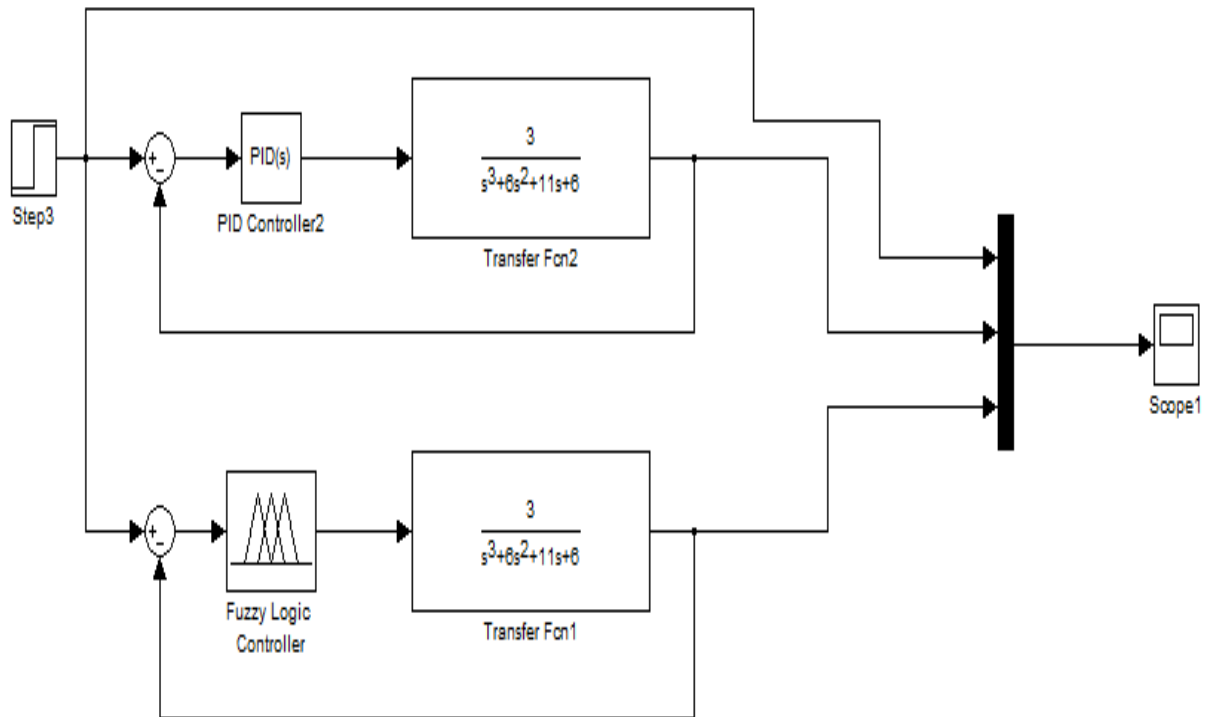


Fig. III.35 – Schéma bloc de système du 3^{ème} ordre commandé par le PID et un contrôleur flou

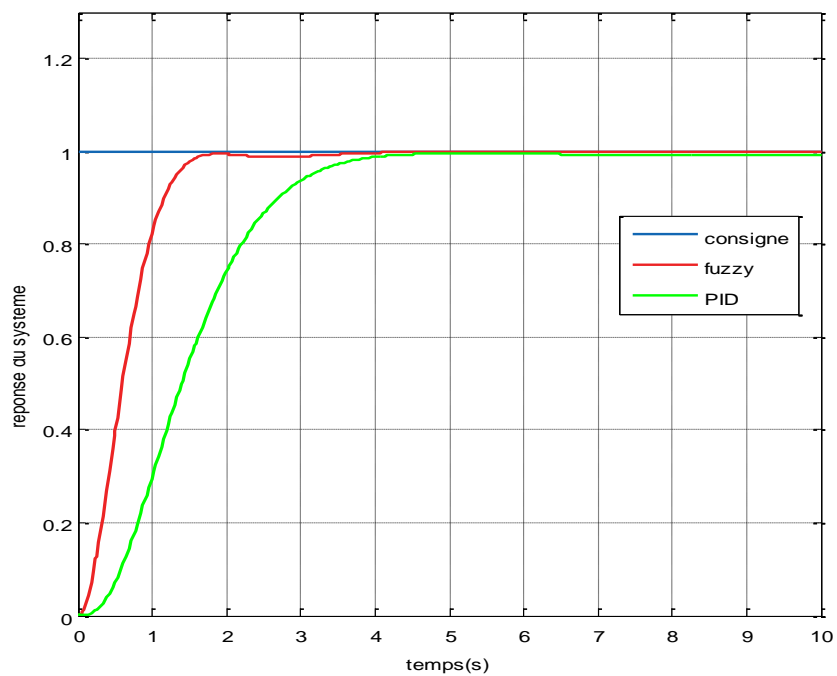


Fig. III.36 – La réponse du système du 3^{ème} ordre.

En observant les résultats précédents, on voit que le comportement des deux régulateurs est semblable du côté de stabilité dans le régime permanent, mais différent en prenant les autres performances. Le régulateur Flou présente par rapport au PID :

- ✓ Un meilleur temps de montée ou temps de réponse.
- ✓ Absence de dépassement.
- ✓ Une bonne précision.

III.5 Conclusion

Dans ce chapitre, on a appliqué deux méthodes de régulation : la commande PID et la commande floue sur plusieurs systèmes avec et sans perturbation afin de pouvoir faire une étude comparative entre ces deux commandes. Nous avons fait des études comparatives en simulation entre le correcteur PID et la commande floue pour réguler des systèmes de différents ordres (1^{er}, 2^{ème} et 3^{ème} ordre).

Après simulation, les résultats montrent que le régulateur flou donne de meilleures réponses par rapport au régulateur PID à savoir un temps de montée plus court, une bonne précision, absence de dépassements. Malgré les avantages du régulateur flou liés à sa mise en œuvre en comparaison avec le PID mais ce dernier reste encore le plus utilisé dans l'industrie.

CONCLUSION GENERALE

Conclusion générale

Le travail présenté dans ce mémoire concerne l'application des deux méthodes de régulation qui sont la commande PID et la commande floue sur plusieurs systèmes avec et sans perturbation afin de pouvoir faire une étude comparative entre ces deux commandes et voir les avantages et les inconvénients de chacune d'entre elles.

Le premier chapitre a permis de rappeler les généralités sur les systèmes asservis et les différentes performances d'un système ainsi que les systèmes élémentaires. Après le concept fondamental de la commande classique conventionnelle à base des régulateurs PID est présenté. Nous avons cité ensuite les différents types du correcteur P, PI, PD et PID, leur principes de fonctionnement en boucle ouverte et en boucle fermée, aussi que les différentes structures d'un régulateur PID (série, parallèle et mixte). Après, nous avons cité les différentes méthodes de synthèse du régulateur PID puis on a détaillé le principe de la méthode de Ziegler –Nichols qui permet de calculer de façon systématique les paramètres du régulateur PID.

Dans le deuxième chapitre, nous avons présenté les notions de base de la théorie de la logique floue, ainsi que la commande floue. Après, nous avons illustré la structure élémentaire d'un contrôleur flou et on a précisé le rôle de chaque bloc (base des règles, moteur d'inférences floues, fuzzification, défuzzification) ainsi que les deux types de régulateurs flous les plus connus Takagi-sugeno et Mamdani.

Le dernier chapitre est divisé en trois parties : la première partie est consacrée pour la régulation d'un système de 1^{er} ordre présentée par une serre où on a bien précisé la régulation de la température à base du correcteur PID avec perturbation, après on a fait la commande

PID avec et sans perturbation d'un système de 2^{ème} ordre illustrée par un moteur à courant continue, puis la même procédure précédente pour un système de 3^{ème} ordre. Dans la deuxième partie nous avons donné une démonstration des étapes à suivre pour la création d'un contrôleur flou dans MATLAB-Simulink, puis la simulation des trois systèmes étudiés précédemment avec un régulateur flou de type Mamdani. La dernière partie a été consacrée pour la comparaison en simulation des deux types de régulateur PID et flou, en se basant sur la qualité d'asservissement et les performances des trois systèmes après la régulation.

les résultats ont montré que le régulateur flou donne de meilleures réponses par rapport au régulateur PID à savoir un temps de montée plus court, une bonne précision, absence de dépassements. Malgré les avantages du régulateur flou liés à sa mise en œuvre en comparaison avec le PID mais ce dernier reste encore le plus utilisé dans l'industrie.

A la lumière des résultats obtenus, il serait intéressant dans l'avenir d'aborder les thèmes suivants :

- Combiner entre le flou et le conventionnel pour commander des systèmes non linéaires.
- Appliquer le réglage flou pour contrôler des systèmes réels tels que l'humidité et l'éclairage de la serre pour le système de 1^{er} ordre par exemple.
- Faire une réalisation de l'équipement et du contrôleur flou au niveau d'une serre à Jijel ou d'un moteur à courant continu.

BIBLIOGRAPHIE

Bibliographie

- [1] Maurice Rivoire J-LF, Ed. Eyrolles. Cours d'automatique, tome 2 – Asservissement, régulation, commande analogique.
- [2] Philippe de Larminat EH. Automatique: Commande des systèmes linéaires.
- [3] Eric Magarotto, "cours de régulation", université de Caen, France.
- [4] Jean pierre Corriou, « commande des procédé », edition Lavoisier, Paris 1996.
- [5] Boukhris M. Regulation de vitesse d'une machine asynchrone, Université de Kasdi Merbah OURAGLA, Master en Electrotechnique (2014)
- [6] Vrancic DYR. Anti-windup and conditionnel Transfer for PID controller Conference Anti-windup and conditionnel Transfer for PID controller vol. Vol .A. p. pp 183-6.
- [7] Christophe Le Lann, 2007, Le PID utilisé en régulation de position et/ou de vitesse de moteurs électriques, <http://www.totofweb.net/projets/pid/rapport.pdf>
- [8] Vrancic D, Y.Reng, J.Petrovêiv, and R.Hanns. A new tuning method for PID controllers, 4th IFAC conference on system structure and control, Bucharest, october 23-25, 1997
- [9] Ben Kherif O, Etude comparative en simulation entre un régulateur PID classique et un régulateur PID Intelligent, master en Electronique, Université MOHAMED BOUDIAF de M'SILA, 2016
- [10] Zoghmar M, Habchi A, Étude comparative entre deux régulateurs PID et FLC appliqués à la Machine à Courant Continu, master Génie électrique, Université LARBI BEN M'HIDI DE OUM ELBOUAGHI, 2012
- [11] A. Zaatri, 2015, 'Asservissements et Régulation Industrielle, Cours destiné aux élèves de 4^{ème} Année Ingénieurs, Université Mentouri, Constantine.
- [12] Zadeh LA. Fuzzy sets. Information and control. 1965;8(3):338-53.
- [13] Tong-Tong J-R. La logique floue: Hermes, 1995.

- [14] Kaufmann A. Introduction à la théorie des sous-ensembles flous à l'usage des ingénieurs: Éléments théoriques de base: Masson, 1973.
- [15] Jang J-SR, Sun C-T, Mizutani E. Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE Transactions on automatic control. 1997;42(10):1482-4.
- [16] Zadeh LA. A fuzzy-algorithmic approach to the definition of complex or imprecise concepts. International Journal of Man-machine studies. 1976;8(3):249-91.
- [17] HUERTA EB. Logique floue et algorithmes génétiques pour le pre-traitement de données de biopuces et la sélection de gènes: Université d'Angers, 2008.
- [18] S A. théorie des ensembles flous: Application à la mesure de la pauvreté au congo 2009.
- [19] Soufane H, Halim r. Optimisation de la commande de l'ivitation magnétique pour la recherche locale. Mester en électronique, université Mohammed Seddik Ben Yahia-Jijel. 2017.
- [20] Belhouane S. Étude et simulation d'une régulation thermique par la logique floue Master en Electronique, Université ABDERRAHMANE MIRA de Bejaia,(2014).
- [21] Talbi N. Conception des Systèmes d'Inférence Floue par des approches Hybrides : Application pour la Commande et la Modélisation des Systèmes Nonlinéaires. Thèse de Doctorat en Science, Université de Constantine 1. (2014).
- [22] Ghaleb A. Commande floue d'un UPFC à trois niveaux pour améliorer la stabilité transitoire d'un réseau multimachines: mémoire de magistère en électrotechnique, université des sciences et de la technologie «Mohamed Boudiaf »ORAN,2012.
- [23] F.C. Teng, A.C. Tsoi, "Fuzzy controller based on MacVicar Whelan matrix", 20th International conference on Industrial Electronics, Control and Instrumentation, 1994.

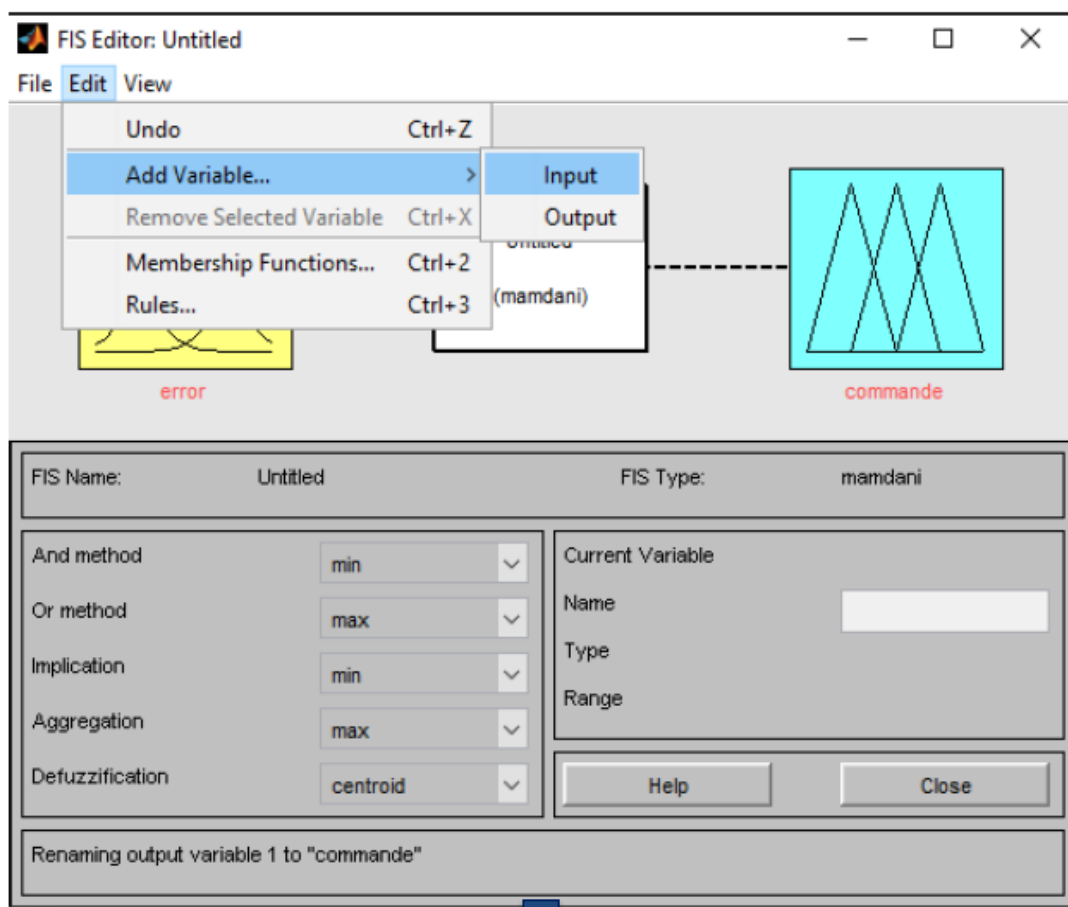
ANNEXE

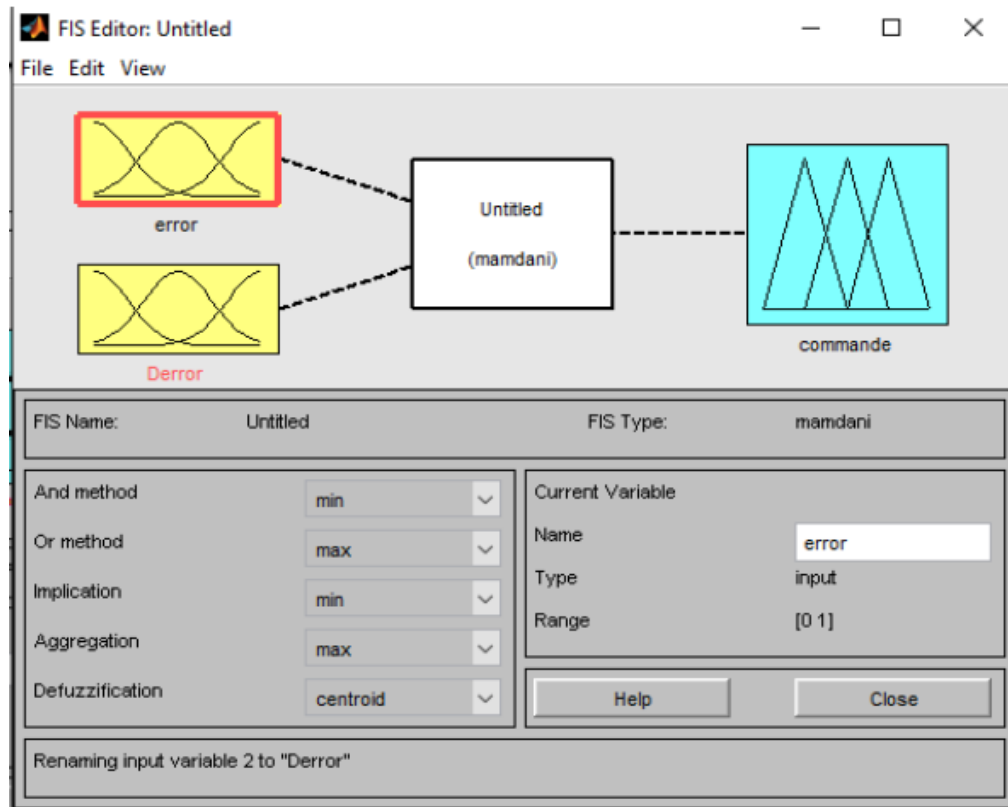
Étapes pour créer un régulateur flou deux entrées en utilisant la boîte à outil fuzzy-logic :

Pour configurer le contrôleur flou à deux entrées dans le Simulink de MATLAB, il faut passer par les étapes suivantes :

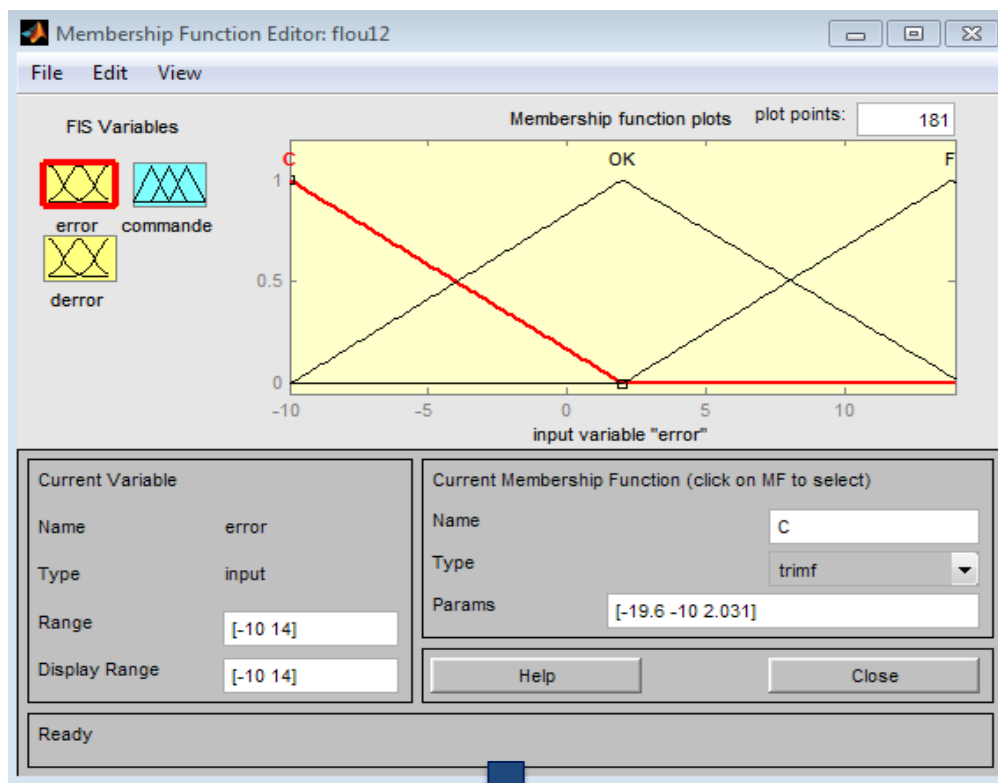
- On commence par ouvrir *Matab Command*, et taper >>fuzzy, la boîte à outil de « fuzzy logic » va s'ouvrir. Après on suit les étapes suivantes :
- **Edit Add input**

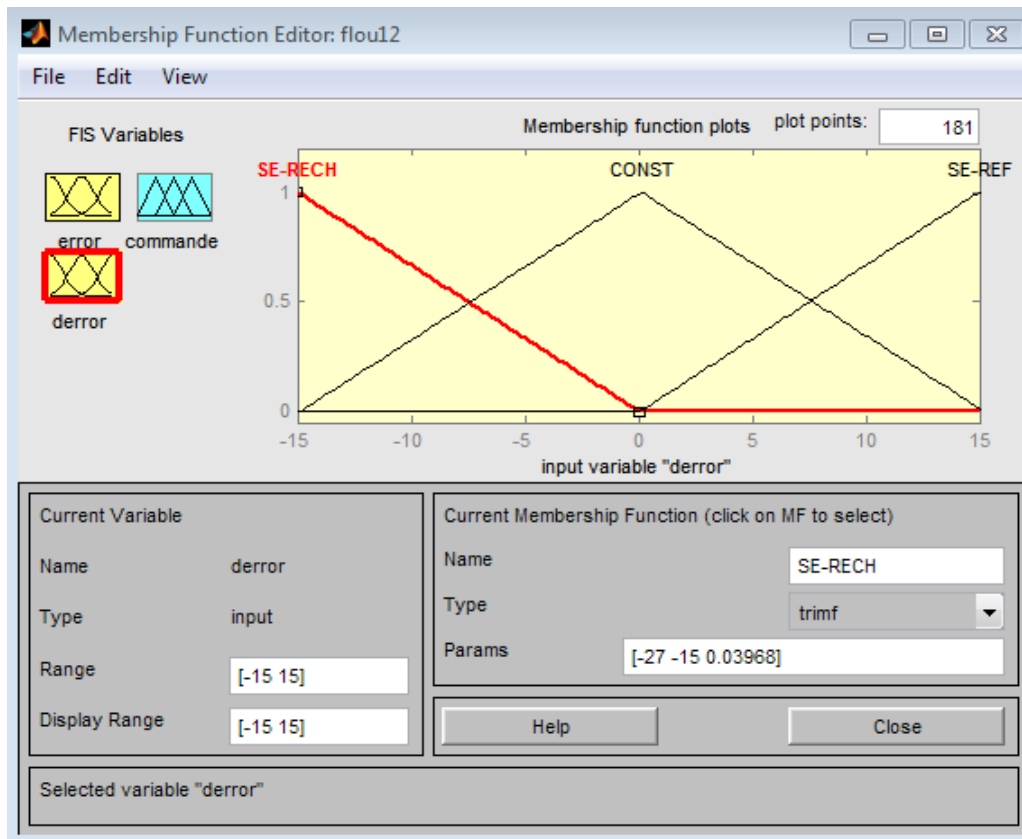
On commence tout d'abord par définir les entrées et les sorties, et par la suite on va essayer de lier les fonctions d'appartenance d'une façon logique.



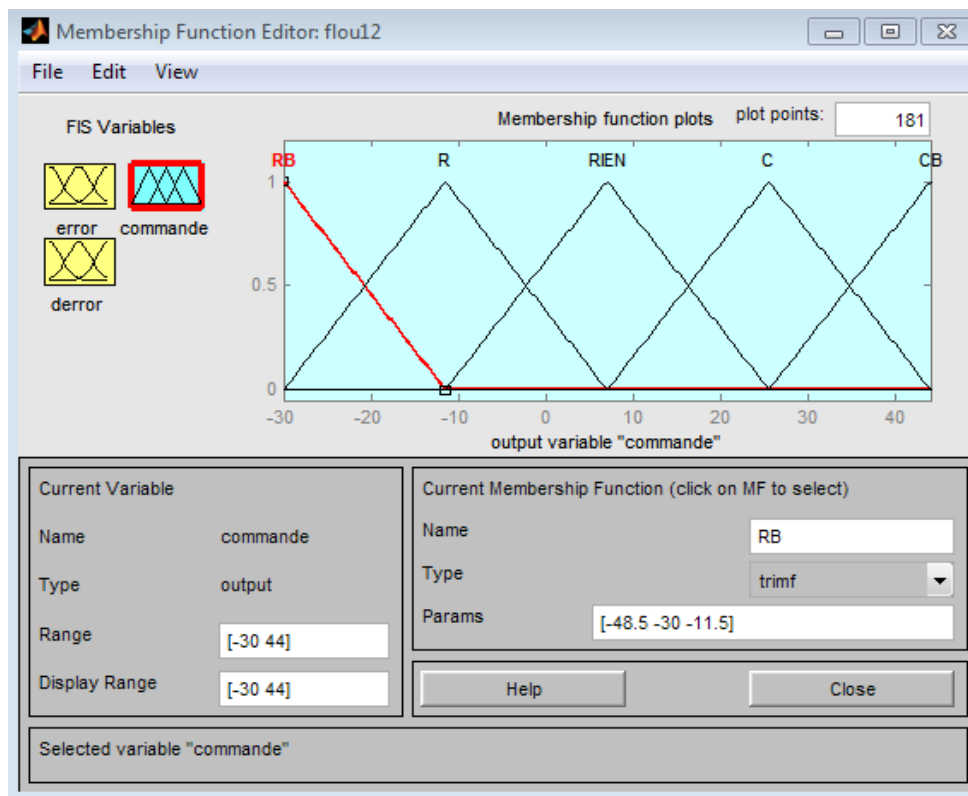


- **View Edit Membership function**
- **Edit Add Mfs** (pour les entrées :erreur et la dérivée de l'erreur)

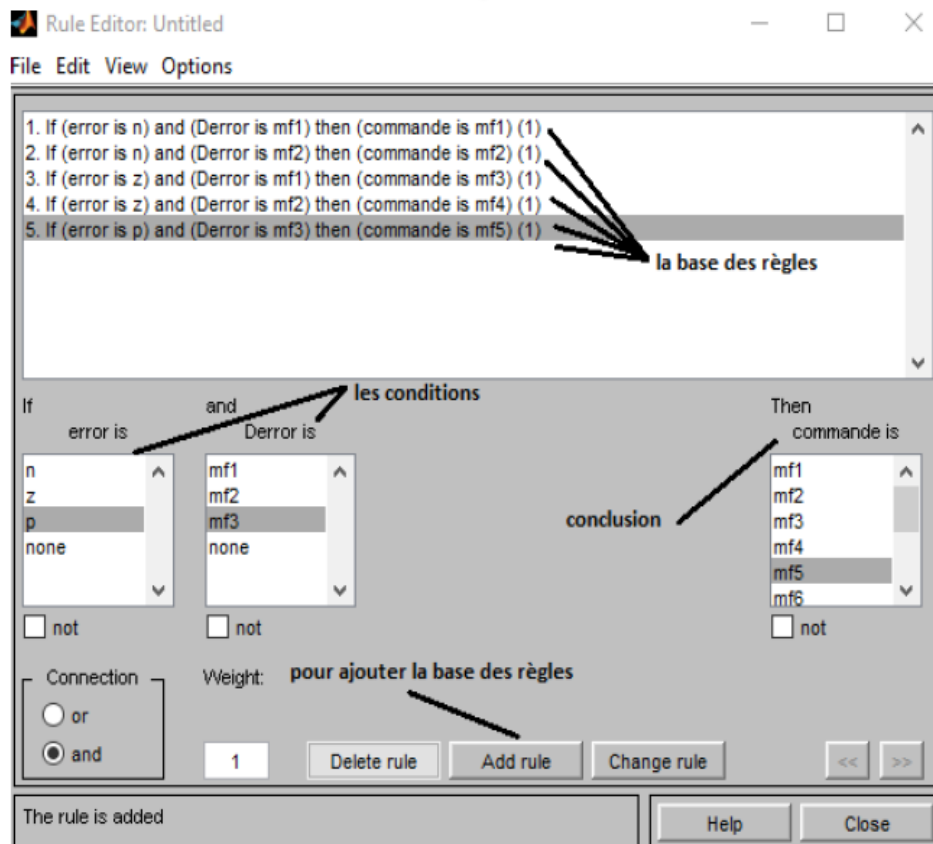
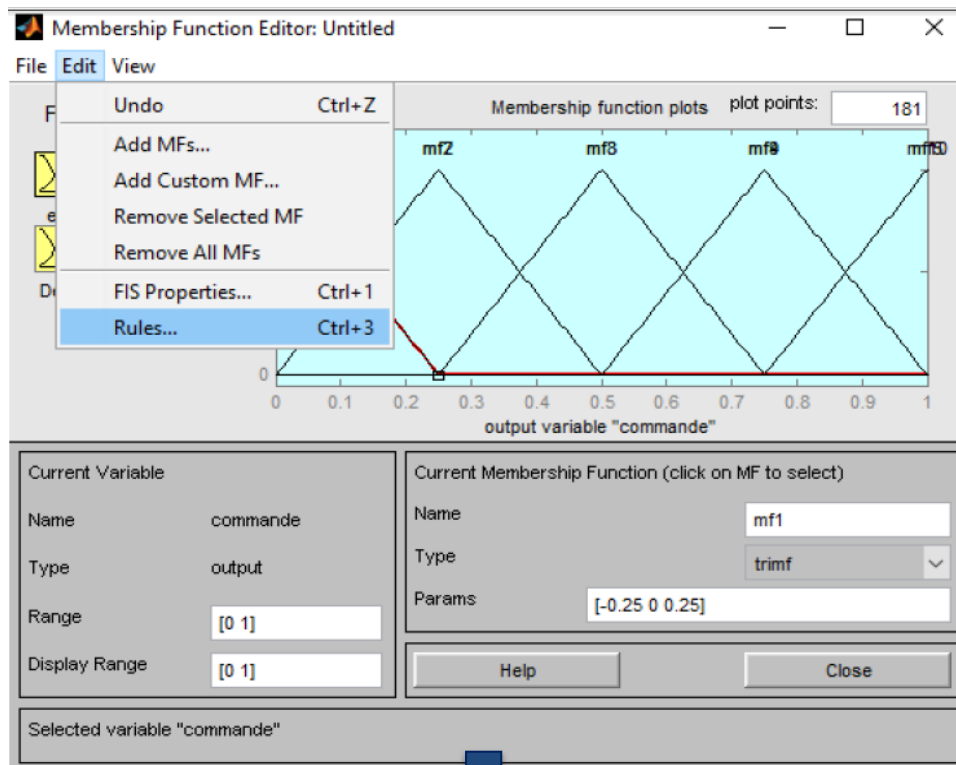




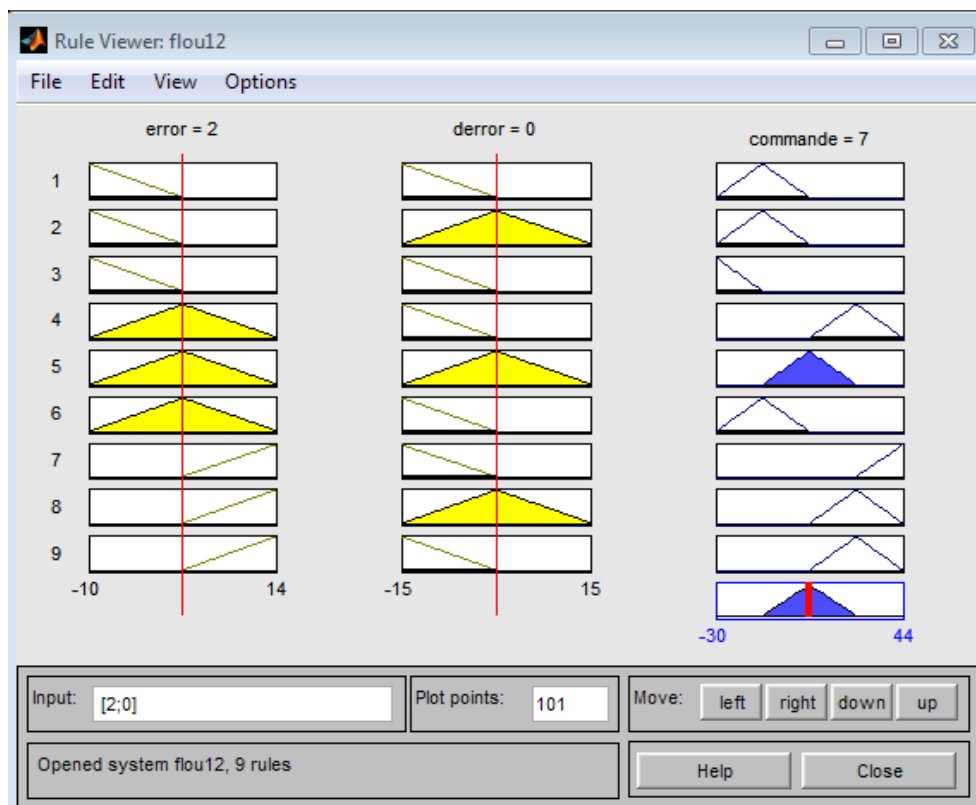
- View Edit Membership function
- Edit Add Mfs (pour la sortie :commande)



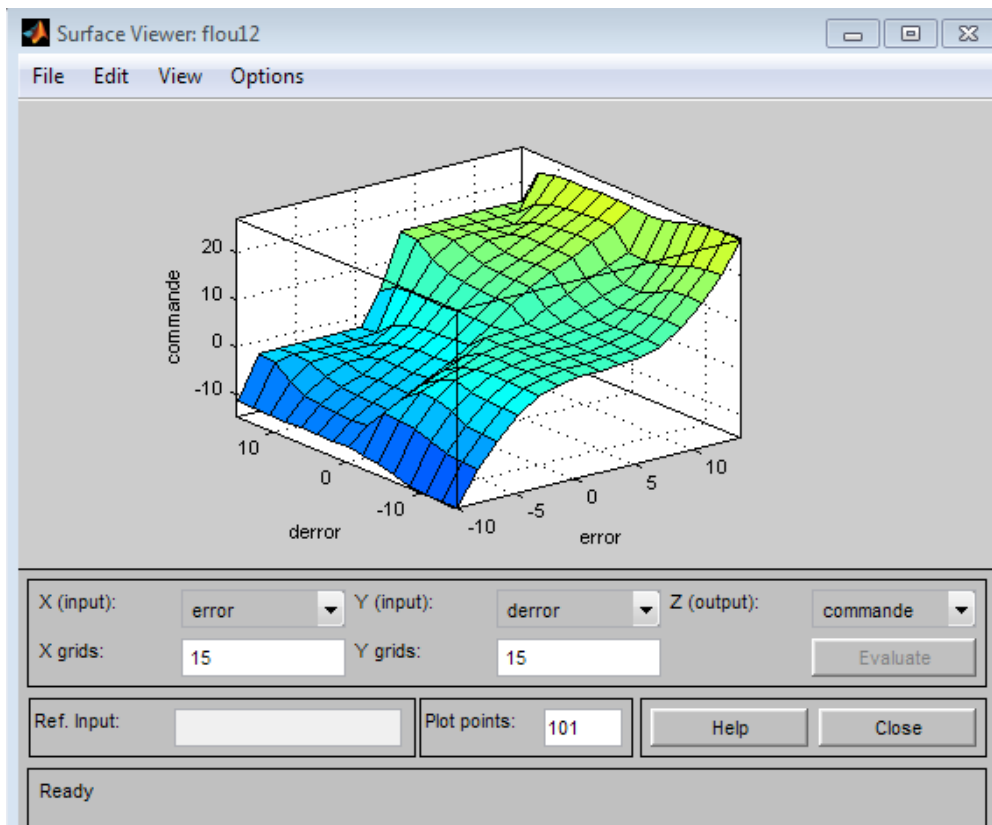
➤ View Edit rule



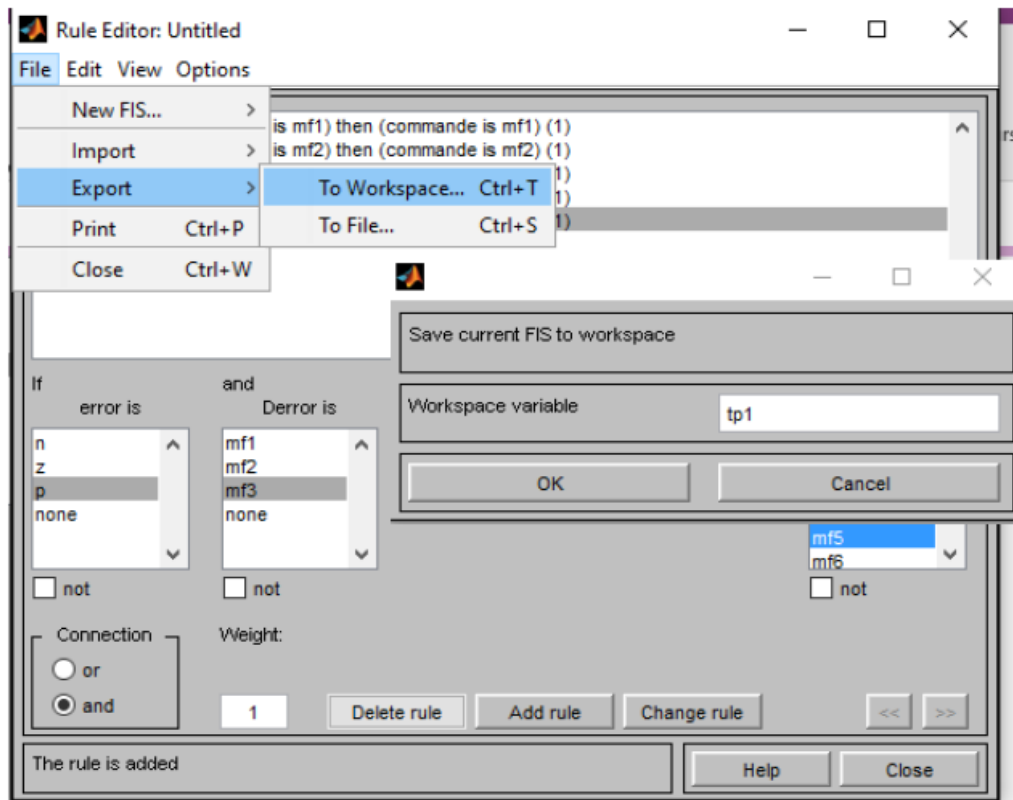
➤ **View view rule**



➤ **View view surface**



➤ Enregistrement dans workspace Matlab



➤ Compiler le fichier “.fis” dans le bloc “fuzzy logic controller”

