

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université 8Mai 1945 – Guelma
Faculté des Sciences et de la Technologie
Département d'Electronique et Télécommunications



Mémoire de Fin d'Etude

Pour l'obtention du Diplôme de Master Académique

Domaine : Sciences et Technologie
Filière : Electronique
Spécialité : Systèmes Electroniques

**Adaptation des paramètres d'un contrôleur PID par la
logique floue**

Présenté par :

BOUDRA Mohammed Amin

Sous la direction de :

Dr. Nemissi Mohamed

Juin 2017

Dédicace



Je dédie ce mémoire à mes parents pour leur amour inestimable, leur confiance, leur soutien, leur sacrifices et tous les valeurs qu'il ont su m'inculquer.

A mes frères ainsi qu'à mes sœurs pour leur tendresse, leur complicité et leur présence malgré la distance qui nous sépare.

A mon superviseur Mr. Nemissi Mohamed

A tout ma famille ainsi qu'à mes amis.

Et à tout.

Boudra M^d Amin

Remerciement

J'adresse mes remerciements aux personnes qui m'ont aidé dans la réalisation de ce mémoire.

En premier lieu, je remercie Mr. Nemissi Mohamed, docteur à l'université de Guelma, pour ses précieux conseils et ses encouragements lors de la réalisation de ce mémoire.

Je tiens ensuite à remercier mes parents pour le soutien inconditionnel dont ils ont fait preuve depuis que mon projet professionnel est défini. Merci pour le soutien financier, moral et psychologique. Si je suis ici aujourd'hui, c'est grâce à vous !

Je remercie également toutes les personnes qui, de près ou de loin, ont participé à l'élaboration de ce mémoire.

Enfin, je remercie mes amis et camarades de promotion pour ces années passées ensemble, dans les meilleurs moments comme dans les pires.

Boudra M^{ed} Amin.

Résumé

Dans ce travail, nous étudions une méthode d'adaptation des paramètres d'un contrôleur PID. Il s'agit d'appliquer un système flou en ligne pour déterminer les paramètres du contrôleur en fonction du signal d'erreur et de sa première dérivée. Les paramètres réglant le fonctionnement du contrôleur flou, comme les fonctions d'appartenance et les règles floues sont introduites par un expert humain.

TABLE DES MATIERES

RESUME :	iii
TABLE DES MATIERES	v
INTRODUCTION GENERALE	1
CHAPITER I : Concepts de base de la logique floue	2
I.1. INTRODUCTION	3
I.2. HISTORIQUE	3
I.3. DOMAINES D'APLICACION	4
I.4. THEORIE DES ENSEMBLES FLOUS	5
I.5. LES FONCTIONS D'APPARTENANCE LES PLUS UTILISENT	5
I.6. CARACTERISTIQUES D'UN ENSEMBLES FLOUS	6
I.6.1 Support	7
I.6.2 La Hauteur	7
I.6.3 Noyau	7
I.7. LES OPERATIONS SUR LES ENSEMBLES FLOUS	8
I.7.1 Complément (L'opérateur Non)	8
I.7.2 Union (L'opérateur OU)	9
I.7.3 Intersection (L'opérateur ET)	10
I.8. IMPLICATIONS FLOUE	12
CHAPITER II : LA COMMANDE FLOUE	14
II.1. INTRODUCTION AUX METHODES DE COMMANDE	15
II.2. QUELQUES PREMIERE REELES APLICACION DE LA LOGIQUE FLOUUE	15
II.2.1 Les Machlne à laver floue	15
II.2.2 Stabilisation d'image numérique	16
II.2.3 Les Systèmes flous dans les voitures	16
II.2.4 Système floue dans le métro	17
II.3. LES SYSTEMES D'INFERENCE FLOUS	18
II.3.1 La base de règle	19
II.3.2 Les étapes d'un système d'inférence floue	20
II.3.2.1 Fuzzification	20
II.3.2.2 Inférence floue	20
a. La méthode des inférences	20
b. La méthode d'inférence max-min	21
c. La méthode d'inférence max-pro	21
d. La méthode d'inférence somme-pro	21
II.3.2.3. Defizzufication	21
a. La méthode du maximum	21
b. La défuzzification par centre de gravité	22
II.4. SYSTEME DE MAMDANI	23
II.5. SYSTEME DE SUGENO ET TAKAGI	24

CHAPITER III : ADAPTATION FLOU DES PARAMETRES D'UN CONTROLEUR PID	26
III.1. INTRODUCTION	27
III.2 LE REGULATEUR PID	27
III.2.1 Structure d'un régulateur PID	27
III.2.2 Réglage statique des gains d'un PID	28
III.2.3 Indices de performance d'un PID	29
III.3. REGLAGE DYNAMIQUE DES GAINS D'UN PID	30
III.4. ADAPTATION FLOUE DES PARAMETTRE D'UN PID	31
III.5. RESULTATS	34
III.6. CONCLUSION	39
CONCLUSION GENERALE	40
BIBLIOGRAPHIE	41

Introductions générale

La régulation automatique des processus est omniprésente dans presque tous les systèmes, tant dans le domaine scientifique que technologique. Les systèmes actuels deviennent plus en plus complexes et les performances des régulateurs utilisés ne cessent de s'améliorer. Les méthodes de réglage conventionnelles comme la commande optimale, la commande adaptative ou la commande robuste, se basent sur une connaissance plus ou moins précise du modèle mathématique du système à réguler. Lorsque le système est fortement non-linéaire, imprécis ou très complexe, il est parfois impossible de définir un modèle mathématique de son fonctionnement. C'est ainsi que des méthodes telles que la logique floue et les réseaux de neurones constituent une voie prometteuse pour aborder ces problèmes de régulation. Ces méthodes ont de grandes capacités de traitement des données non-linéaires.

Dans ce travail, nous utilisons un système flou afin de régler dynamiquement les gains d'un régulateur PID. Cette méthode permet de faire intervenir dans le contrôleur des connaissances acquises par un expert humain. L'intérêt principal de cette méthode est qu'elle ne nécessite pas de connaître le fonctionnement du système, mais simplement la façon de le commander. Les connaissances sont exploitées sous une forme linguistique par l'intermédiaire de règles comme « si condition alors action ». Une méthode de raisonnement (inférence) utilise ces règles pour définir les commandes qui sont envoyées au système.

Le présent mémoire se compose de trois chapitres :

Le premier chapitre constitue une introduction à la logique floue. Il présente le cadre général de la logique floue, son historique et ses concepts de bases comme les sous-ensembles flous et les opérateurs flous.

Le deuxième chapitre concerne la commande floue. Il décrit les systèmes d'inférences floues et leurs différentes étapes en mettant l'accent sur les deux systèmes les plus utilisés Mamdani et de Sugeno.

Dans le dernier chapitre nous appliquons un système flou à base de règles pour l'adaptation des paramètres d'un régulateur PID. Nous évaluons les performances de ce système en le comparant avec un régulateur PID ayant des valeurs initialisées par la méthode de Ziegler-Nichols.

Chapitre I : Concept de base de la logique floue

I.1. Introduction

Dans le raisonnement humain de nombreux concepts ne sont pas assez précis dans le sens d'être complètement vrai ou faux, plutôt ils peuvent être interprétés d'une manière plus qualitative. Dans la vie quotidienne, nous utilisons des concepts comme grand, petit, rapide, lent, bon, mauvais.... qui sont difficiles à traduire numériquement. La logique et l'inférence classiques résonnent avec ces concepts de façon naturelle dans sa vie quotidienne, notre quête de connaissances scientifiques nous a conduits à aborder le problème de la représentation de ces concepts d'une manière plus systématique et plus précise. C'était Zadeh qui a formalisé pour la première fois le concept du flou, et plus tard le domaine de la logique floue a suscité l'intérêt d'une grande part de la communauté scientifique, la théorie des ensemble flous et la logique floue ont été appliquées dans presque tous les domaines.

I.2. Historique

En 1965, le professeur Lotfi A. Zadeh, de l'université de Berkeley (Californie), a déclaré : < un contrôleur électromécanique doté d'un raisonnement humain serait performant qu'un contrôleur classique >. Il a publié un article où il introduit le terme < fuzzy > dans la littérature technique et donne naissance au concept d'ensemble flou. Le professeur tentait de modéliser les phénomènes sous forme floue afin de remédier aux limitations dues aux incertitudes des modèles classiques à équation différentiel. Il a réalisé depuis, des avancées majeures dans le domaine et a été rapidement accompagné par de nombreux chercheurs développant des travaux théoriques.

En 1974, le docteur Mamdani, de l'université de Londres, a introduit la commande floue dans la régulation d'un processus industriel, en réalisant un contrôleur flou expérimental sur un moteur à vapeur. Le problème sur qu'il s'était penché, avait la réputation d'être difficile et les résultats obtenus étaient très encourageants. En 1978, la société danoise F.L. Smith & Co a réalisé la première véritable application industrielle de la logique floue : le contrôle d'un four à ciment.

A partir des années 80, les recherches se sont arrêtées en Europe et ont été reprises au Japon où plusieurs applications ont commencé à émerger.

En 1985, la logique floue a été introduite au Japon par le chercheur T.Sugeno. Les sociétés japonaises ont très vite compris son avantage à la fois technique et commercial. La recherche n'était pas uniquement théorique mais également appliquée. A la fin des années 80, ce fut une véritable explosion du flou. Les Japonais ont mis sur le marché plus de 2000 produits flous destinés au grand public, notamment des appareils électroménagers et du matériel audio-visuel. Dans l'industrie, l'application de la logique floue s'est étendue au traitement des eaux, aux grues portuaires, aux métros, aux systèmes de ventilation et de climatisation ainsi qu'un autre domaine très différent tel que la finance ou le diagnostic médical.

A partir de 1990, apparaissent de nombreuses applications en Allemagne et à moindre échelle aux Etats unis d'Amérique.

De nos jours, une intense activité de recherche a débuté afin de combiner la logique floue aux algorithmes génétiques et aux réseaux de neurones.

I.3. Domaine d'application

Les applications pratiques déjà connues sont innombrables, nous ont citons quelques-unes :

Contrôle :

- Aviation (Rockwell)
- Métro (Hitachi)
- Boîte de vitesses (Nissan, Subaru)
- Amarrage de navettes spatiales (NASA)

Ordonnements :

- Ascenseurs (Mitsubishi, ...)

Traitement du signal :

- Reconnaissance de caractères (Palm Sanyo)
- Autofocus (Sanyo, Canon)
- Stabilisateur d'images (Panasonic)

La logique floue comporte des applications dans d'autre domaine telle que la recherche opérationnelle, la gestion financière, le contrôle de production, la robotique, la météorologie, l'informatique, la médecine, l'intelligence artificielle, la neuro mimétique, la linguistique, le traitement des données, la communication homme-machine, la reconnaissance des formes, la fiabilité, la chimie industrielle...

Aujourd'hui, nous avons beaucoup de possibilité de travailler avec des logiciels qui comporte la logique floue :

- C++ : "fuzzy-lite", "jFuzzyQt".
- Java : "jFuzzyLogic".
- MATLAB : "Fuzzy Logic Toolbox " (utilisé dans ce mémoire).

I.4. Théorie des ensembles flous

Dans la théorie des ensembles classiques, l'appartenance d'un élément à un sous-ensemble est définie par une valeur logique standard: 1 si l'élément appartient au sous-ensemble, 0 sinon. Dans la théorie floue, un élément peut appartenir en partie à un sous-ensemble : son degré d'appartenance est décrit par une valeur comprise entre 0 et 1.

Par exemple, la logique classique peut facilement partitionner la température d'une pièce en deux sous-ensembles, «moins de 15 degrés» et «15 degrés ou plus». Toutes les températures de moins de 15 degrés sont alors considérées comme appartenant à l'ensemble «moins de 15 degrés». On leur affecte une valeur de 1. Toutes les températures atteignant 15 degrés ou plus ne sont pas considérées comme appartenant à l'ensemble «moins de 15 degrés». On leur attribue une valeur de 0. Cependant, le raisonnement humain s'appuie fréquemment sur des connaissances ou des données inexactes, incertaines ou imprécises. Une personne placée dans une pièce dont la température est soit de 14.95 degrés soit de 15.05 degrés, ne fera certainement pas de distinction entre ces deux valeurs. Cette personne sera pourtant capable de dire si la pièce est «froide» ou «chaude», sans pour cela utiliser de température limite ni de mesure précise.

I.5. Les fonctions d'appartenance les plus utilisées

La logique floue est basée sur la notion d'appartenance partielle ; un élément peut alors appartenir partiellement à plusieurs sous-ensembles. Un sous-ensemble flou B est défini par une fonction d'appartenance μ_B qui associe à chaque élément x un degré $\mu_B(x)$ compris entre 0 (non-appartenance) et 1 (appartenance totale)

La logique floue possède des règles de calcul bien précises permettant de combiner plusieurs fonctions d'appartenance et de tirer des conclusions pondérées :

Fonction d'appartenance trapézoïdale :

$$u(x, a, b, d) = \max\left(0, \min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right)\right) \quad (1.1)$$

Fonction d'appartenance triangulaire :

$$u(x, a, b, d) = \max\left(0, \min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right)\right) \quad (1.2)$$

Fonction d'appartenance gaussienne :

$$f(x) = e^{-\frac{(x-u)^2}{2\sigma^2}} \quad (1.3)$$

Fonction d'appartenance cloche généralisée :

$$f(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (1.4)$$

Fonction d'appartenance sigmoïdale :

$$f(x) = \frac{1}{1 + e^{-a(x-c)}} \quad (1.5)$$

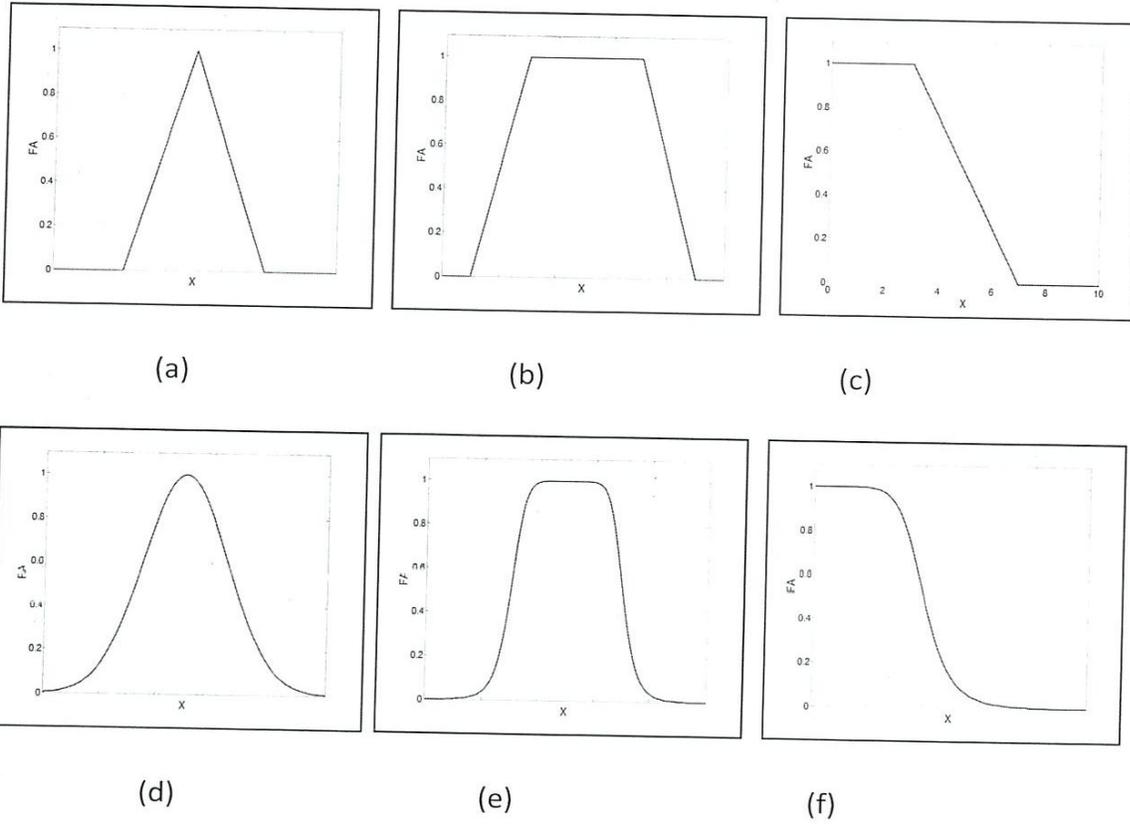


Fig I.1. Exemples de fonction d'appartenance

- | | |
|---------------------------------|---------------------------|
| (a).Fonction triangulaire | (b).Fonction trapézoïdale |
| (c).Fonction linéaire saturable | (d).Fonction gaussienne |
| (e).Fonction cloche généralisée | (f).Fonction sigmoïdale |

I.6. Caractéristiques d'un sous-ensemble flou

Un sous-ensemble flou est complètement défini par la donnée de sa fonction d'appartenance. A partir d'une telle fonction, un certain nombre de caractéristiques du sous-ensemble flou peuvent être étudiées. [1]

I.6.1. Support

Le support d'un sous-ensemble flou de X , noté $Supp(A)$, est l'ensemble de tous les éléments qui lui appartiennent au moins un petit peu. Formellement:

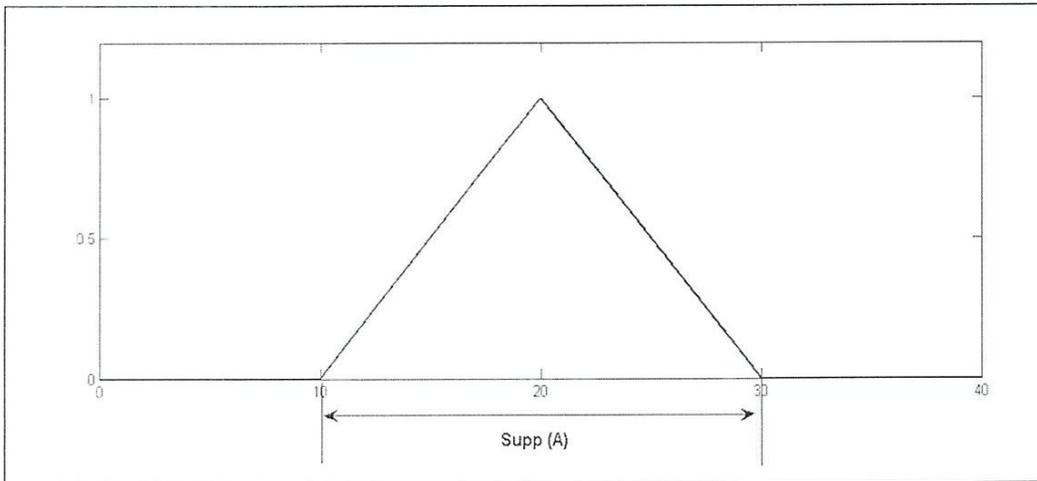


Fig I.2. Le support du sous-ensemble flou

I.6.2. La Hauteur

La hauteur du sous-ensemble flou A de X , notée $h(A)$, est le plus fort degré avec lequel un élément de X appartient à A . Formellement:

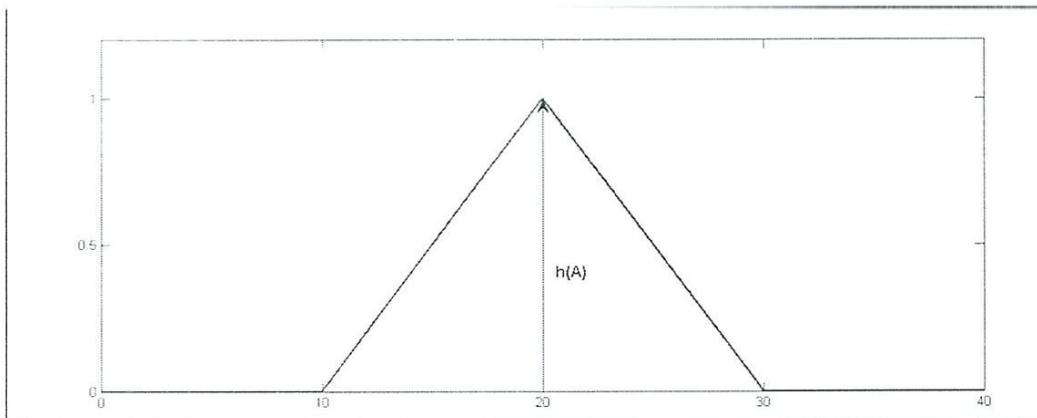


Fig I.3. La hauteur du sous-ensemble flou

I.6.3. Noyau

Un sous-ensemble flou est normalisé si sa hauteur $h(A) = 1$. Le noyau d'un sous-ensemble flou A de X , noté $Noy(A)$, est l'ensemble de tous les éléments qui lui appartiennent totalement (avec un degré 1). Formellement :

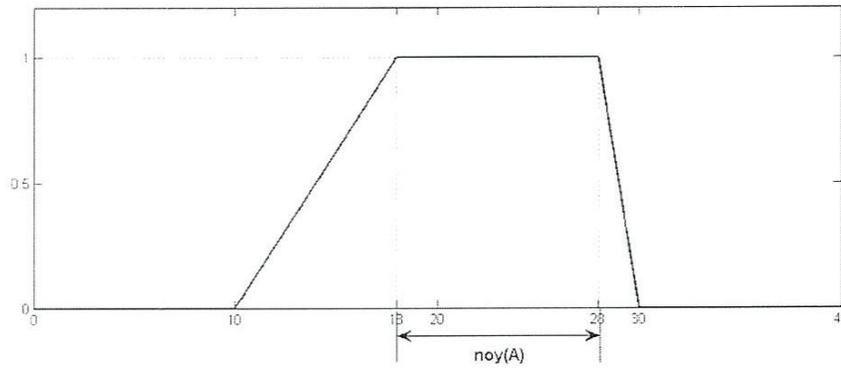


Fig.I.4. Le noyau d'un sous-ensemble flou

1.7. Les opérations sur les ensembles flous

Les règles d'inférences font appel aux opérateurs « et », « ou » et « non », qui s'appliquent aux variables floues. Dans le cas de la logique binaire ces opérateurs sont définis de façon simple et univoque. Dans le cas de la logique floue, la définition de ces opérateurs n'est plus univoque.[2]

1.7.1. Complément (L'opérateur Non)

La fonction de membre du complément d'un ensemble flou B avec la fonction d'appartenance $\mu(A)$ est définie comme la négation de la fonction d'appartenance spécifiée. Ceci est appelé le critère de négation (figure I.5). L'opérateur « non » est alors donné par :

$$\mu(B) = 1 - \mu(A) \quad (1.6)$$

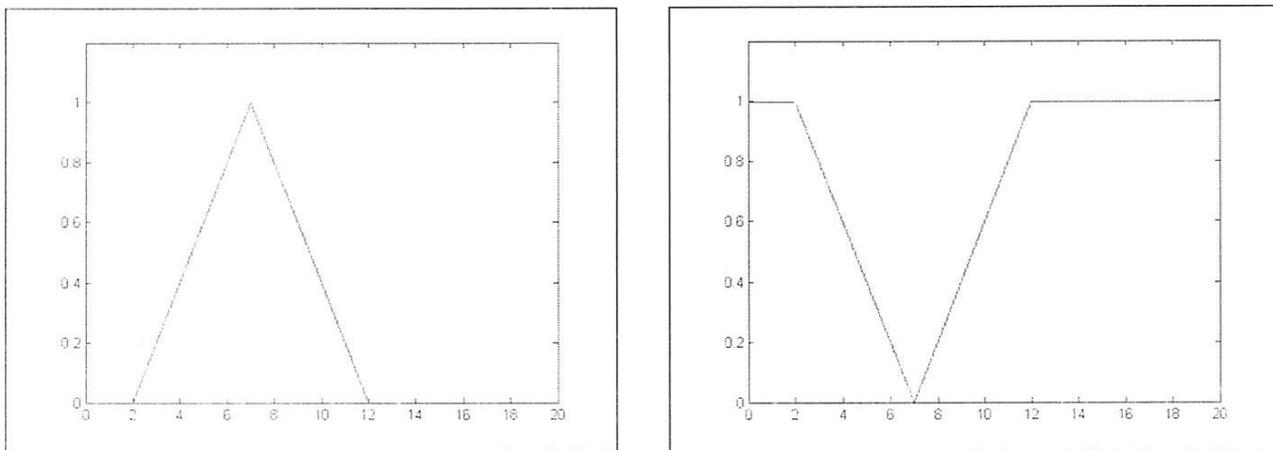


Fig I.5. L'opérateur NON de sous-ensemble flou A.

I.7.2. Union (L'opérateur OU)

La fonction de membre de l'Union des deux ensembles flous A et B avec des fonctions d'appartenance μ_A et μ_B respectivement est définie comme le maximum des deux fonctions individuelles d'adhésion. Ceci est appelé le critère maximal.

$$\mu_{A \cup B} = \max(\mu_A, \mu_B) \quad (1.7)$$

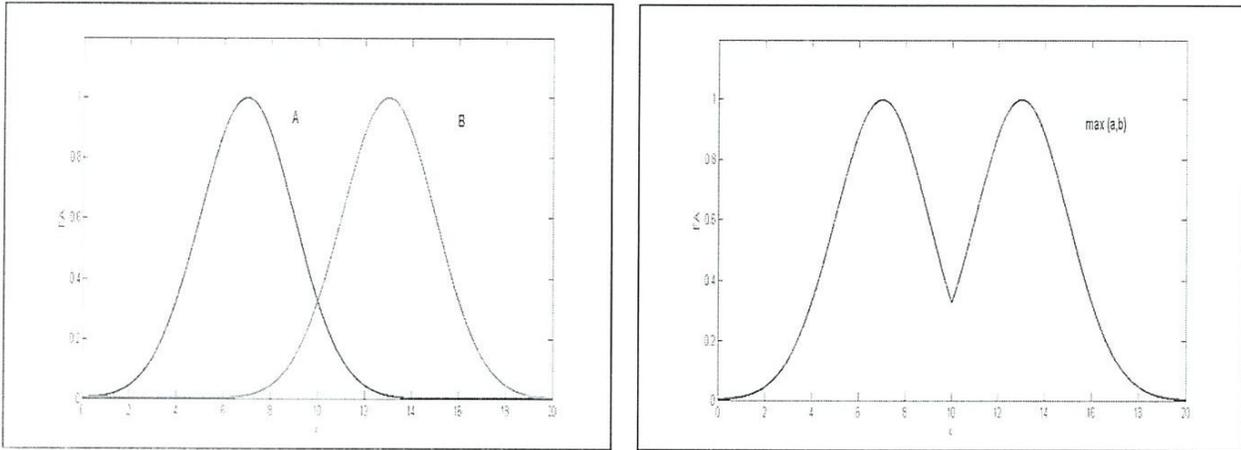


Fig I.6. L'opérateur OU de deux sous-ensemble flou A et B.

La figure (1.7) illustre un exemple de l'opérateur OU avec la méthode de la somme algébrique

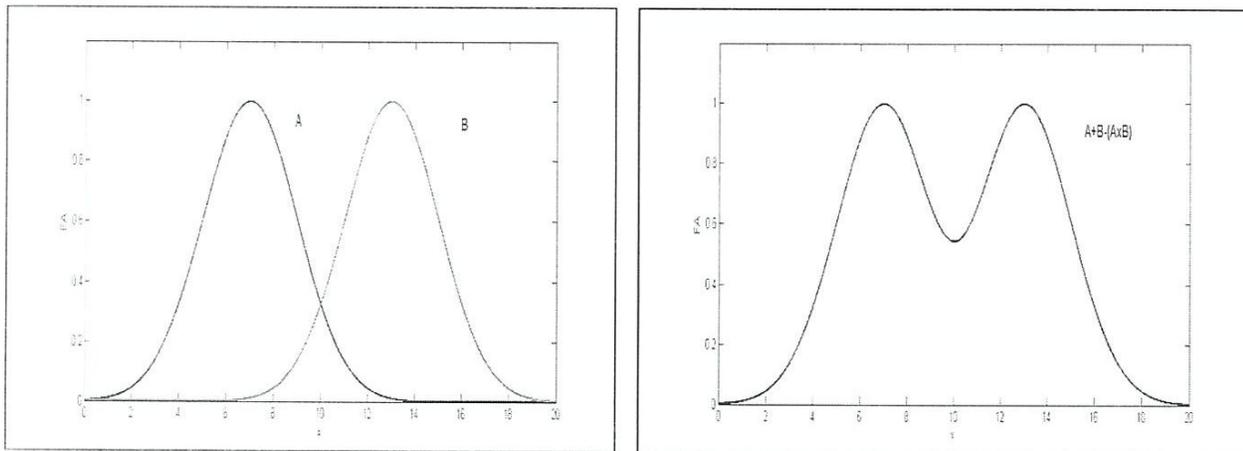


Fig I.7. L'opérateur OU de deux sous-ensemble flou A et B.

Il existe de nombreuses possibilités pour représenter l'opérateur OU. Les plus répandus sont [3]:

t-conorme	nom
$\max(x, y)$	Zadeh
$x + y - x \cdot y$	Probabiliste
$\min(x + y, 1)$	Lukasiewicz
$\frac{x + y - xy - (1 - \delta)xy}{1 - (1 - \delta)xy}$	Hamacher($\delta > 0$)
$\begin{cases} x & \text{si } y = 0 \\ y & \text{si } x = 0 \\ 1 & \text{sinon} \end{cases}$	Weber

Tableau I.1. Les t-normes les plus utilisées

I.7.3. Intersection (L'opérateur ET)

La fonction de membre de l'intersection de deux ensembles flous A et B avec des fonctions d'appartenance u_A et u_B respectivement est généralement définies comme le minimum des deux fonctions individuelles d'adhésion. Ceci est appelé le critère minimum (figure I.8).

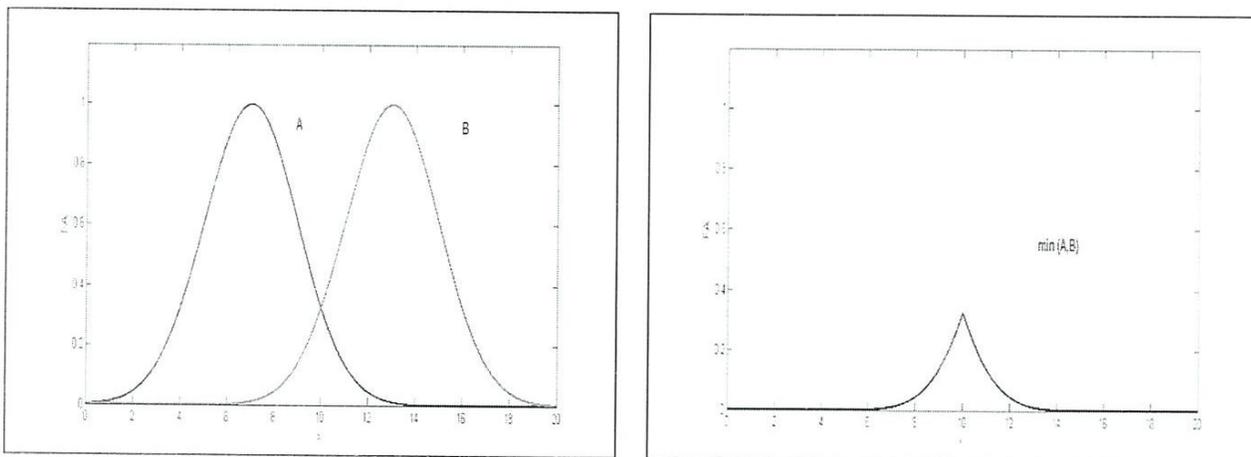


Fig I.8. L'opérateur ET de deux sous-ensembles flous A et B

La figure (1.9) illustre un exemple de l'opérateur ET avec la méthode de la produit algébrique

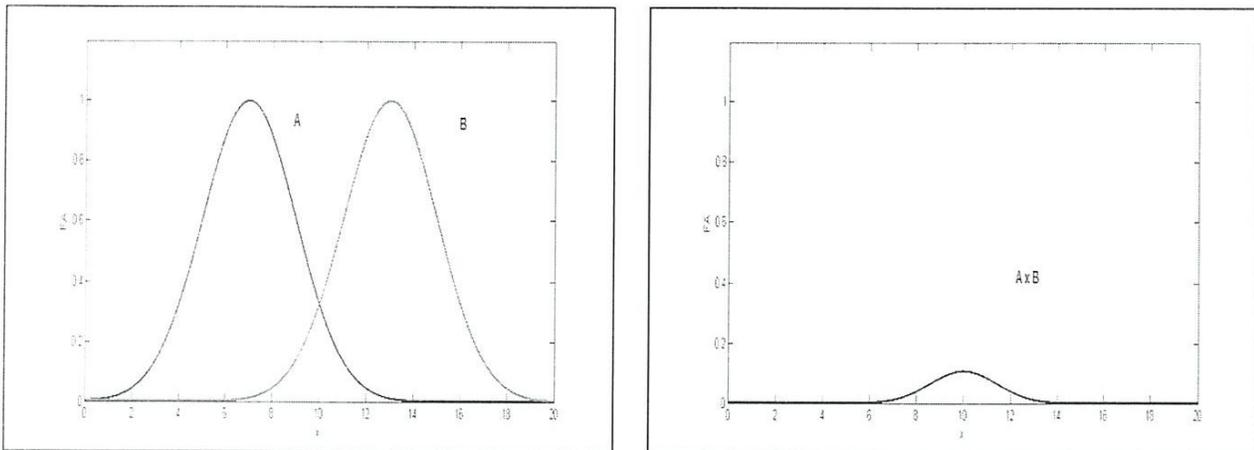


Fig 1.9. L'opérateur ET de deux sous-ensemble flou A et B

Cet opérateur peut être réalisé de plusieurs façons. Cependant les solutions les plus simples et les plus utilisées pour caractériser la satisfaction simultanée de deux propriétés sont [3] :

t-norme	nom
$\min(x, y)$	Zadeh
$x \cdot y$	Probabiliste
$\max(x + y - 1, 0)$	Lukasiewicz
$\frac{xy}{\delta + (1 - \delta)(x + y - xy)}$	Hamacher ($\delta > 0$)
$\begin{cases} x & \text{si } y = 1 \\ y & \text{si } x = 1 \\ 0 & \text{sinon} \end{cases}$	Weber

Tableau I.2. Les t-conorme les plus utilisées

I.8. Implication floue

Dans la proposition floue p : « Si (X est A) ; alors (Y est B) », les propositions « X est A » et « Y est B » sont construiront des deux variables linguistiques $(x, T(x), X, G, M)$ et $(y, T(y), Y, G, M)$ qui sont a priori indépendantes. L'implication floue permet de définir une liaison entre la prémisse « X est A » et la conclusion « Y est B » de cette règle.

Définition : considérons p : « X est A », q : « Y est B » deux propositions floues construites à partir de deux variables linguistiques $(x, T(x), X, G, M)$ et $(y, T(y), Y, G, M)$. Notons a, b les valeurs de vérité possibles de p et q respectivement.

Une implication floue, que l'on note (a, b) , est une fonction :

$$I(a, b): [0,1] \times [0,1] \rightarrow [0,1] \quad (1.8)$$

qui définit la valeur de vérité de la proposition floue « si p alors q ». L'implication floue est également notée :

$$A \rightarrow B \quad (1.9)$$

D'après la définition, les valeurs de vérité sont exprimées par :

$$a = \mu_A(x),$$

$$b = \mu_B(y),$$

La fonction $I(a, b)$ est donc équivalente à la fonction d'appartenance, que l'on note $\mu_R(x, y)$, d'une relation floue définie entre X et Y . cette fonction d'appartenance s'exprime, pour tout (x, y) de X et Y , en utilisant la fonction d'appartenance $\mu_A(x)$ intervenant dans la prémisse et celle $\mu_B(x)$ intervenant dans la conclusion de la règle :

$$\mu_R(x, y) = I(\mu_A(x), \mu_B(x)) \quad (1.10)$$

Ainsi, la définition d'une implication floue peut se ramener à celle d'une fonction d'appartenance. Les implications floues la plus souvent employées sont précisées dans le tableau (I.3)

Parmi les implications définies dans le tableau, les implications de Mamdani et Larsen sont les plus connues en raison de leurs applications dans la commande floue

Valeur de vérité	$I(\mu_A(x), \mu_B(x))$	nom
I_m	$\min(\mu_A(x), \mu_B(x))$	Mamdani
I_l	$\mu_A(x) \times \mu_B(x)$	Larsen
I_r	$1 - \mu_A(x) + \mu_A(x) \times \mu_B(x)$	Reichenbach
I_w	$\max(1 - \mu_A(x), \min(\mu_A(x), \mu_B(x)))$	Willmott
I_{rg}	$\begin{cases} 1 & \text{si } \mu_A(x) \leq \mu_B(x) \\ 0 & \text{sinon,} \end{cases}$	Rescher-Gaines
I_{kd}	$\max(1 - \mu_A(x), \mu_B(x))$	Kleene-Dienes
I_{bg}	$\begin{cases} 1 & \text{si } \mu_A(x) \leq \mu_B(x) \\ \mu_B(x) & \text{sinon} \end{cases}$	Brouwer-Gödel
I_g	$\begin{cases} \min\left(\frac{\mu_B(x)}{\mu_A(x)}, 1\right) & \text{si } \mu_A(x) \neq 0 \\ 0 & \text{sinon} \end{cases}$	Goguen
I_l	$\min(1 - \mu_A(x) + \mu_B(x), 1)$	Lukasiewicz

Tableau I.3. Implications floues les plus utilisées

Chapitre II :

La Commande floue

II.1. Introduction aux méthodes de commande:

De façon générale, un système a pour objectif de piloter l'entrée d'un processus afin d'obtenir un fonctionnement correct de ce dernier. Lorsqu'on dispose d'un modèle plus ou moins précis du système à commander, on peut utiliser un contrôleur de structure standard, fixe ou adaptatif, dont les paramètres seront évalués à partir du modèle. Malheureusement, lorsque le système est difficilement modélisable, la conception du contrôleur peut s'avérer très complexe, sinon impossible.

Lorsqu'un opérateur humain commande manuellement un système, les actions qu'il réalise sont dictées par une connaissance *subjective* du fonctionnement de ce système. Par exemple, s'il fait « froid » dans une pièce, on « augmente » le chauffage ; s'il fait « très froid » on « chauffe plus ». Cette commande du système peut être envisagée de façon différente selon la personne qui la réalise : la sensation de « froid » n'est pas directement liée à une mesure de la température.

Ce principe est à la base de la commande floue. La mesure réalisée sur le système (« température ») est prise en compte par l'intermédiaire d'une variable linguistique (« froid », « tiède », « chaud »), qui est issue d'une analyse par un expert humain. Ensuite, l'action à réaliser est déduite à la fois d'un ensemble de règles de commande (« s'il fait froid, on chauffe plus ».....) et de l'état du système, qualifié par la variable linguistique. Enfin, la commande finale du système est créée en utilisant les conclusions de la déduction.

En résumé, un contrôleur flou comporte les différents éléments suivants :

- Un sous-système d'interface avec le flou, composé en général d'un ensemble de variables linguistiques
- Une base de connaissances : « base de données » et « base de règles linguistique de commande ».
- Un sous-système réalisant un raisonnement en utilisant des méthodes issues de la logique floue.
- Un sous-système d'interface avec le non flou, qui fournit la ou les commandes envoyées au système.

II.2. Quelques premières réelles applications de la logique floue :

Dans ce paragraphe nous citons quelques applications réelles de la logique floue. Il s'agit de quelques réalisations très importantes dans les années 90 [4] ;

II.2.1. Les machines à laver floues

Les machines à laver floues ont été les premiers grands produits de consommation à utiliser des systèmes flous. Ils ont été produits par « Matsushita Electric Industrial Company » au Japon vers 1990. Ils utilisent un système flou pour régler automatiquement le cycle approprié en fonction du type et de la quantité de saleté et de la taille de la charge. Plus

précisément, le système flou utilisé est un système à trois entrées et une sortie, où les trois entrées sont des mesures de la saleté, du type de saleté et de la mesure de la charge, et la sortie est le cycle correct. Les capteurs fournissent les entrées au système flou. Le capteur optique envoie un faisceau de lumière à travers l'eau et mesure le temps dans lequel il atteint l'autre côté. Plus l'eau est sale, plus la lumière prend du temps. Le capteur optique peut également indiquer si la saleté est boueuse ou huileuse. La saleté boueuse se dissout plus rapidement. Donc, si les lectures de lumière atteignent le minimum rapidement, la saleté est boueuse. Si la baisse est plus lente, il est huileux. Et si la courbe incline quelque part entre les deux, la saleté est mélangée. La machine dispose également d'un capteur de charge qui enregistre le volume de vêtements. De toute évidence, plus le volume des vêtements est important, plus il faut de temps de lavage. Les heuristiques ci-dessus ont été résumées dans un certain nombre de règles Si-Alors floues qui ont ensuite été utilisées pour construire le système flou.

II.2.2. Stabilisateur d'image numérique

N'importe qui a déjà utilisé un caméscope se rend compte qu'il est très difficile pour une main humaine de tenir le caméscope sans le secouer légèrement et en conférant un tremblement ennuyeux à la bande. La résolution de tel problème produirait une nouvelle génération de caméscopes ayant une énorme valeur commerciale. « Matsushita » a présenté ce qu'il appelle un stabilisateur d'image numérique, basé sur des systèmes flous, ce qui stabilise l'image lorsque la main tremble. Le stabilisateur d'image numérique est un système flou qui est construit en fonction des règles suivantes :

Si tous les points de l'image se déplacent dans la même direction, Alors la main tremble

Si seulement certains points de l'image se déplacent, Alors la main ne tremble pas

Plus précisément, le stabilisateur compare chaque image actuelle avec les images précédentes en mémoire. Si le tout semble avoir décalé, alors selon la première règle précédente la main tremble et le système flou permet de régler le cadre pour compenser. Sinon, cela le laisse seul. Ainsi, si une voiture traverse le champ, seule une partie de l'image changera, de sorte que le caméscope n'essaye pas de compenser. De cette façon, l'image reste stable, bien que la main tremble

II.2.3. Les systèmes flous dans les voitures

Une automobile est une collection de nombreux systèmes - moteur, transmission, frein, suspension, direction et autres- et les systèmes flous ont été appliqués à presque tous ces systèmes. Par exemple, Nissan a breveté une transmission automatique floue qui économise du carburant de 12 à 17 pour cent. Il est basé sur l'observation suivante. Un changement de vitesse normal se produit à chaque fois que la voiture passe une certaine vitesse, elle change donc assez souvent et chaque changement consomme du carburant. Cependant, les conducteurs humains ne changent pas moins fréquemment, mais tiennent également

compte d'autres facteurs. Le dispositif de transmission automatique flou de Nissan résume ces heuristiques dans une collection de règles Si-Alors floues qui ont ensuite été utilisées pour construire un système flou pour guider les changements de vitesse.

Nissan a également développé un système de freinage antiblocage flou. Le défi ici est d'appliquer la plus grande pression du frein sans causer un blocage. Le système Nissan considère un certain nombre d'heuristiques, par exemple :

Si la voiture ralentit très rapidement, Alors le système suppose un blocage et diminue la pression.

En avril 1992, « Mitsubishi » a annoncé un système omnibus flou qui contrôle la transmission automatique, la suspension, la traction, la direction à quatre roues d'une voiture, la transmission à quatre roues motrices et le climatiseur. La boîte à vitesse floue dégrade dans les virages et empêche également la voiture de se déplacer de manière inappropriée sur les virages ou lorsque le conducteur lâche l'accélérateur. La suspension floue contient des capteurs à l'avant de la voiture qui enregistrent des vibrations et des changements de hauteur dans la route et ajuste la suspension pour une conduite plus fluide. La traction floue empêche l'excès de vitesse sur les coins et améliore la stabilité sur les routes glissantes en décidant si elles sont au niveau ou en pente. Enfin, la direction floue règle l'angle de réponse des roues arrière en fonction des conditions routières et de la vitesse de la voiture, et la climatisation floue vérifie la lumière du soleil, la température et l'humidité pour améliorer l'environnement à l'intérieur de la voiture.

II.2.4. Systèmes flous dans le métro

L'une des applications les plus importantes des systèmes flous est sûrement le système de contrôle flou pour le métro « Sendai » au Japon. En 1991, ce métro de Sendai avait transporté des passagers pendant quatre ans et était l'un des systèmes de métro les plus avancés. Sur une seule route nord-sud de 13,6 kilomètres et 16 stations, le train fonctionne très bien. Le système de contrôle flou fait quatre critères de performance simultanément : sécurité, confort de conduite, traçabilité à la vitesse cible et précision des arrêts. Le système de commande floue se compose de deux parties : le régulateur de vitesse constant (il démarre le train et maintient la vitesse au-dessous de la limite de sécurité) et le contrôleur d'arrêt automatique (il règle la vitesse du train pour s'arrêter à la position cible). Le régulateur de vitesse constant a été construit à partir de règles telles que :

Pour la sécurité :

Si la vitesse du train approche la vitesse limite, Alors sélectionnez le niveau de freinage maximal.

Pour le confort de la conduite :

Si la vitesse est dans la marge autorisée, Alors ne changez pas de contrôle

D'autres règles ont été utilisées dans le système réel comme le contrôleur d'arrêt automatique qui a été construit selon les règles suivantes:

Pour le confort de conduite :

Si le train s'arrête dans la zone autorisée, Alors ne changez pas le niveau de contrôle

Pour le confort et la sécurité de conduite ;

Si le train est dans la zone autorisée, Alors changez le niveau de contrôle de l'accélération à un léger freinage.

II.3. Les systèmes d'inférence flous

Les systèmes d'inférence flous (SIF), appelés aussi les systèmes experts flous (Fuzzy expert Systems), les modèles flous (Fuzzy Models), ou simplement systèmes flous (Fuzzy Systems), sont des structures de calcul basées sur le concept de la théorie des ensembles flous, des règles si-alors floues et du raisonnement flou. En raison de leur caractère pluridisciplinaire, ils ont été appliqués avec succès dans une grande variété d'applications, tels que l'automatique, la robotique, la reconnaissance des formes ainsi que les systèmes d'aide à la décision.

Un SIF se compose essentiellement de trois éléments conceptuels (figure 2.3): une base de règles qui contient une collection de règles floues, une base de données qui définit les fonctions utilisées dans les règles floues, et un mécanisme de raisonnement qui effectue la procédure d'inférence en se basant sur les règles. Ce processus s'effectue en trois phases.

Fuzzification : Cette étape consiste à déterminer le degré d'appartenance de chaque variable d'entrée aux sous-ensembles flous utilisés dans les prémisses des règles.

Inférence : Il s'agit de la détermination des degrés d'appartenances des règles par la combinaison des propositions de leurs prémisses et du calcul des sorties.

Défuzzification : Le rôle de cette étape est de fournir des valeurs précises des variables de sorties du système afin qu'elle soit exploitable. Par exemple dans le cas de la commande industriel, la sortie du système constitue le signal envoyé au processus.

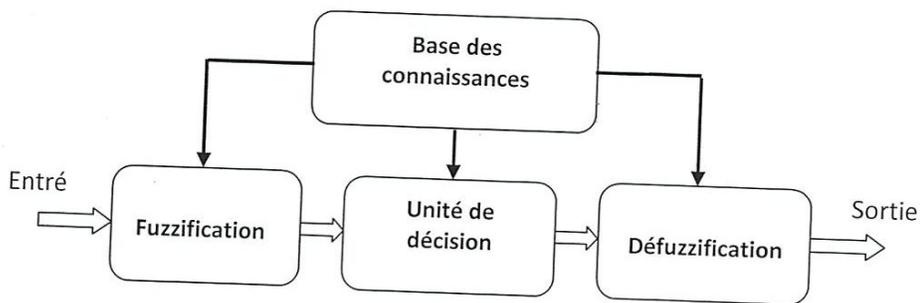


Fig II.1. Système d'Inférence Flou (SIF)

II.3.1. La base de règles

Les règles Si-Alors floues (appelées aussi règles floues) admettent généralement la forme suivante :

$$\text{Si } x \text{ est } A \text{ et } y \text{ est } B \text{ Alors } z \text{ est } C \quad (2.1)$$

Où A, B et C sont des variables linguistiques définies par des sous-ensembles flous.

Une règle floue se compose de deux parties :

La prémisse : formée par une combinaison de prépositions liées entre elles par les opérateurs Et, Ou et Non. Dans l'exemple précédent, la proposition « x est A et y est B » constitue la prémisse de cette règle.

La conclusion : c'est la partie qui définit la conséquence de la règle. Dans la règle précédente, la conclusion est la proposition « z est C ».

La forme de ces règles est très employée dans notre expression quotidienne, par exemple :

$$\text{Si la Résistance est Grande Alors le Courant est Faible} \quad (2.2)$$

Où le Courant et la Résistance sont des variables floues et Faible et Grande sont des termes linguistiques caractérisés par des fonctions d'appartenance.

Une autre forme de règles Si-Alors floues, proposée par Sugeno et Takagi, comporte une conclusion numérique. Dans ces règles, les sous-ensembles flous sont introduits seulement dans les prémisses. Ce type de règles admet la forme :

$$\text{Si } x \text{ est } A \text{ et } y \text{ est } B \text{ Alors } z = f(x, y) \quad (2.3)$$

La fonction $f(x, y)$ est généralement un polynôme de variables d'entrée x et y , mais elle peut être n'importe quelle fonction tant qu'elle permet de décrire convenablement la sortie

du modèle en se basant sur les régions floues définie par les prémisses des règles. Un exemple de telles règles est donné par :

$$\text{Si laFréquence est Grande Alors laTension est 220 V} \quad (2.4)$$

La conclusion (*Tension*) est numérique.

En se basant sur un ensemble de règles floues, un SIF met en œuvre une application non linéaire de l'espace d'entrée à l'espace de sortie. Chacune des règles utilisées décrit le comportement local de cette application ; sa prémisse définit une région de l'espace d'entrée et sa conclusion précise la sortie dans cette région.

II.3.2. Les étapes d'un système d'inférence flou

II.3.2.1. Fuzzification

La fuzzification est réalisée dans l'interface d'entrée du système flou. Cette étape consiste à déterminer le degré d'appartenance de chaque variable d'entrée aux sous-ensembles flous utilisés dans les prémisses des règles.

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

II.3.2.2. Inférence floue

L'inférence floue est le processus de formulation de la relation entre les entrées et les sorties par logique floue. Cette relation offre une base avec laquelle la décision est prise par le système flou. L'inférence floue fait appel alors aux concepts de fonctions d'appartenance, d'implication floue et les règles floues.

$$\underbrace{X \text{ est } A}_{\text{prémisse}} \rightarrow \underbrace{Y \text{ est } B}_{\text{conclusion}} \quad (2.5)$$

a) Les méthodes des inférences

Il existe plusieurs méthodes pour définir les degrés d'appartenance de la variable de sortie ses sous-ensembles flous. Elles se différencient essentiellement par la manière dont vont être réalisés les opérateurs Et et OU utilisés dans les règles d'inférence.

Les trois méthodes d'inférence les plus usuelles sont : Max-min, Max-produit et Somme-produit.

b) La méthode d'inférence max-min :

L'implication floue et l'opérateur ET sont réalisés par la fonction minimum et l'opérateur logique OU est réalisé par la fonction maximum. Elle présente l'inconvénient de nécessiter un temps de calcul trop élevé qui la rend inutilisable pour les applications en temps réel. Cette méthode est également dite "implication de Mamdani". La fonction d'appartenance résultante correspond au maximum des deux fonctions d'appartenance partielles puisque les règles sont liées par l'opérateur OU.

c) La méthode d'inférence max-prod :

La seule différence avec la méthode précédente est la réalisation de l'implication floue par le produit. Les opérateurs logiques ET et OU sont toujours réalisés respectivement par les fonctions minimum et maximum. La méthode max-prod est également dite implication de Larsen.

d) La méthode d'inférence somme-prod :

La méthode d'inférence somme-prod réalise, au niveau de la condition, l'opérateur OU par la fonction somme, et l'opérateur ET par la fonction produit. Au niveau de la conclusion, elle réalise l'implication par la fonction produit.

II.3.2.3. Défuzzification

Le résultat de l'inférence en utilisant une des méthodes d'implication floue, comme Max-min, est une valeur floue. Cette information ne peut être utilisée directement. Une transformation doit être prévue à la sortie du bloc d'inférence pour la convertir en grandeur fixe, cette transformation étant connue par le terme défuzzification (concrétisation). Plusieurs méthodes de défuzzification existent en logique floue, les plus utilisées sont les deux suivant :

a) La méthode du maximum

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

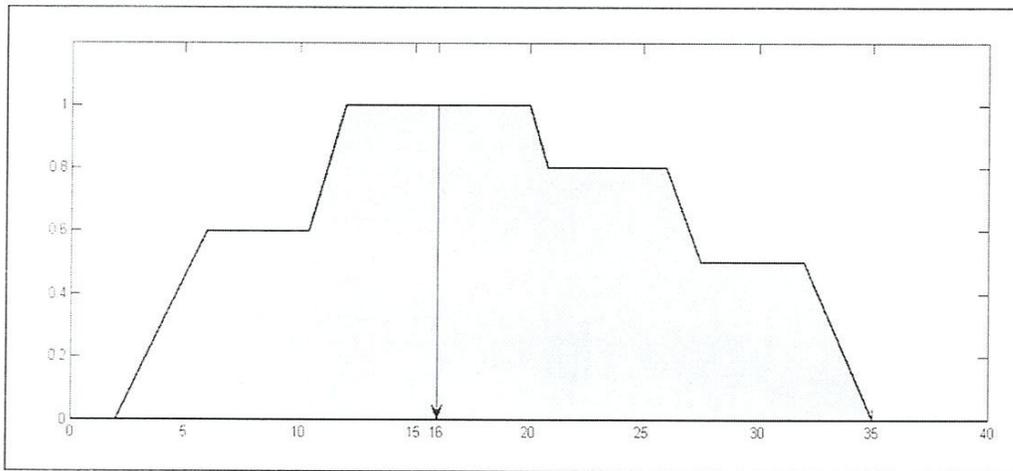


Fig II.2. Défuzzification par la méthode du maximum.

b) La défuzzification par centre de gravité

Il s'agit de calculer le centre de gravité de la fonction d'appartenance de la variable de sortie.

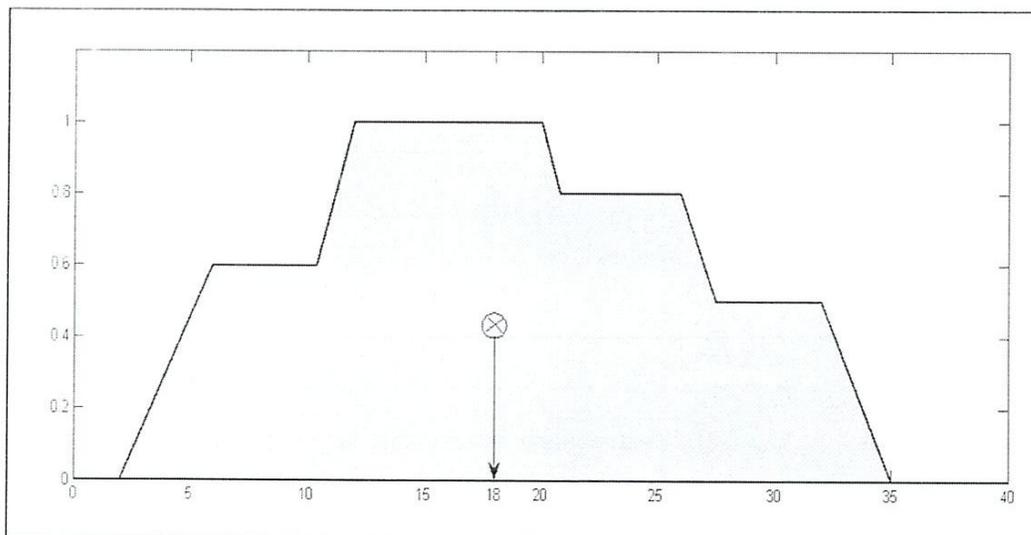


Fig II.3. La défuzzification par centre de gravité

Le calcul du centre de gravité permet bien d'obtenir une seule valeur pour la grandeur de sortie. Son calcul est cependant relativement complexe puisqu'il nécessite le calcul d'une intégrale, ou dans le cas simple de fonctions d'appartenance en raies, d'une somme pondérée.

II.4. Système de Mamdani

Le Système d'Inférence Flou de Mamdani [5] a été proposé en premier lieu comme une tentative de contrôle d'une machine à vapeur et de chaudières. Ce système est basé sur l'utilisation d'un ensemble de règles linguistiques obtenue à partir de l'expérience humaine.

Ce système s'effectue en se basant sur six étapes :

- Détermination d'un ensemble des règles floues.
- Fuzzification des entrées en utilisant les fonctions d'appartenances des entrées correspondantes.
- Combinaison des entrées fuzzifiées correspondantes aux règles floues.
- Détermination des conséquences des règles.
- Combinaison des conséquences pour avoir la distribution de la sortie.
- Défuzzification de la sortie.

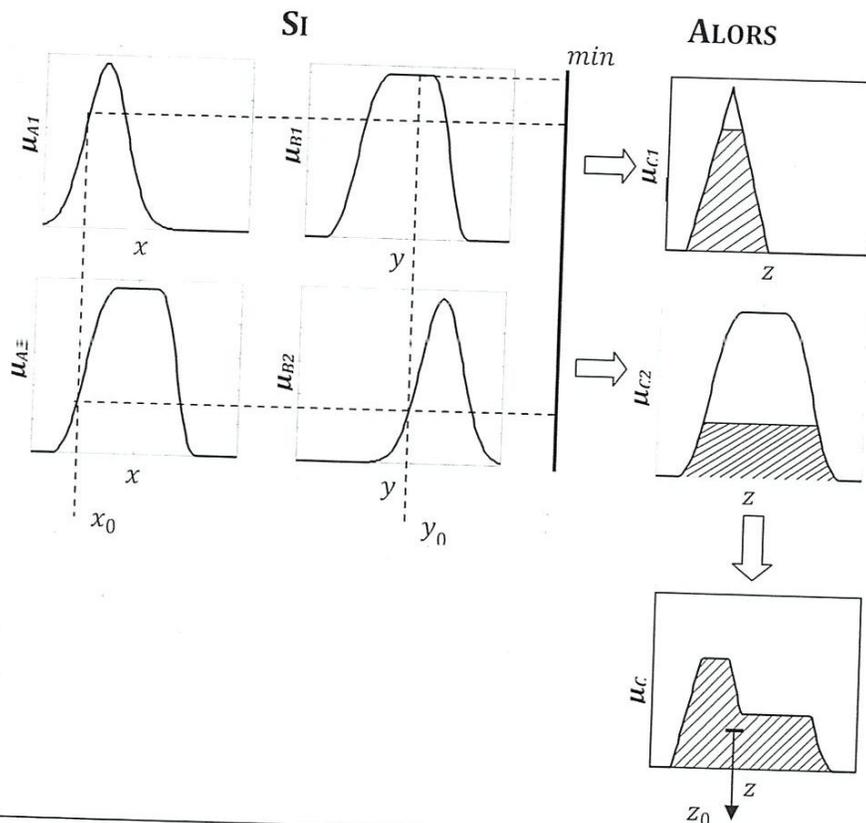


Fig II.4. Système d'inférence de Mamdani

La figure (II.4) illustre un exemple de ce système avec deux variables d'entrée x et y . La fuzzification de la variable d'entrée x s'effectue en utilisant deux sous-ensembles flous A_1 et A_2 , alors que celle de y s'effectue en utilisant B_1 et B_2 .

La sortie est déterminée en calculant le centre de gravité de la distribution de sortie déterminée par la somme des conséquences de règles.

II.5. Système Sugeno et Takagi

Le système de Sugeno et Takagi [6] été proposé afin de concevoir un système de génération des règles floues en se basant sur les entrée/sortie. Ce système emploie des règles de la forme donnée par l'équation 2.5. Il se diffère ainsi du système de Mamdani sur la façon de détermination de la sortie. Le système de Sugeno et Takagi détermine les conséquences des règles par une combinaison linéaire des entrées avec des constantes, la sortie finale sera ensuite calculée par la moyenne pondérée des conséquences des règles.

Un exemple à deux variables d'entrée et deux règles est illustré sur la figure (II.5). Les deux règles sont données par :

$$\begin{cases} \mathcal{R}_1: \text{Si } x \text{ est } A_1 \text{ et } y \text{ est } B_1 \text{ Alors } z_1 = p_1x + q_1y + c_1 \\ \mathcal{R}_2: \text{Si } x \text{ est } A_2 \text{ et } y \text{ est } B_2 \text{ Alors } z_2 = p_2x + q_2y + c_2 \end{cases} \quad (2.6)$$

Les degrés d'activation de ces deux règles sont donnés par :

$$\begin{cases} \alpha_1 = \mu_{A_1}(x_0) \text{ ET } \mu_{B_1}(y_0) \\ \alpha_2 = \mu_{A_2}(x_0) \text{ ET } \mu_{B_2}(y_0) \end{cases} \quad (2.7)$$

Les sorties sont :

$$\begin{cases} z_1 = p_1x + q_1y + c_1 \\ z_2 = p_2x + q_2y + c_2 \end{cases} \quad (2.8)$$

La sortie finale du système est donnée par :

$$z = \frac{\alpha_1 z_1 + \alpha_2 z_2}{\alpha_1 + \alpha_2} \quad (2.9)$$

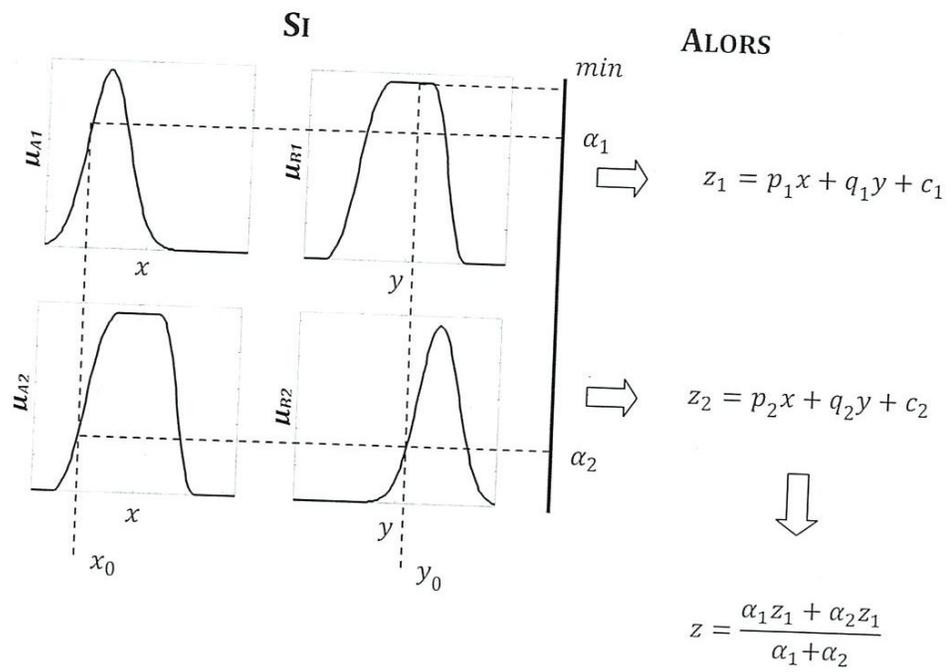


Fig II.5. Système d'inférence de Sugeno et Takagi

Chapitre III: Adaptation floue des paramètres d'un contrôleur PID

III.1. Introduction

Les contrôleurs les plus connus et les plus utilisés dans les processus de contrôle industriel sont des contrôleurs proportionnels-intégral-dérivé (PID) en raison de leurs performances robustes dans une large gamme de conditions de fonctionnement et de simplicité fonctionnelle.

La conception d'un tel contrôleur nécessite une spécification de trois paramètres : gain proportionnel, constante de temps intégrale et constante de temps dérivée. De grand effort a été consacré à développer des méthodes pour réduire le temps consacré à l'optimisation du choix des paramètres du contrôleur.

Les contrôleurs PID dans la littérature peuvent être divisés principalement en deux catégories. Dans la première catégorie, les paramètres du contrôleur sont fixés pendant le contrôle après avoir été réglés ou choisis de manière optimale. La formule de réglage de Ziegler-Nichols est la méthode d'accord la plus connue. Les PID de cette catégorie sont simples, mais ne peuvent pas toujours contrôler efficacement les systèmes avec des paramètres changeants ; ces contrôleurs ont besoin d'une adaptation en ligne de leur paramètres.

Les contrôleurs de la deuxième catégorie ont une structure de contrôleur PID, mais leurs paramètres sont adaptés en ligne ce qui nécessite une certaine connaissance du processus, par exemple La structure du modèle. De tels contrôleurs s'appellent des contrôleurs PID dynamiques ou adaptatifs.

Dans ce travail on s'intéresse à L'application des systèmes à base de connaissance pour l'adaptation des paramètres du régulateur PID. Plus précisément, nous appliquons le système flou à base des règles proposés par Zhao [7].

III.2. Le régulateur PID

III.2.1. Structure d'un régulateur PID

Le régulateur PID est un système linéaire du second ordre à une entrée et une sortie, dont la fonction de transfert dans le domaine de Laplace est donnée par :

$$G(p) = k_p + \frac{k_i}{p} + k_d \cdot p \quad (3.1)$$

Dans laquelle k_p , k_i , k_d sont appelés respectivement gain proportionnel, intégral et dérivé. La commande $c(t)$ du système est ainsi est constituée par la somme pondérée de trois termes dépendant de l'erreur $\varepsilon(p)$ entre la sortie $s(p)$ du processus et la consigne $e(p)$, de la dérivée et de l'intégrale de cette erreur.

On utilise également une autre expression de la fonction de transfert :

$$G(p) = k_p \cdot \left(1 + \frac{1}{T_i \cdot p} + T_d \cdot p\right) \quad (3.2)$$

Où $T_i = K_p/K_i$ est la constante de temps intégrée et $T_d = K_d/K_p$ la constante de temps dérivée.

Dans le domaine temporel, l'évolution de la commande $c(t)$ fournie par le PID est déterminée en fonction de celle de l'erreur $\varepsilon(p)$ en utilisant l'expression suivante :

$$c(t) = K_p \cdot \varepsilon(t) + K_i \cdot \int \varepsilon(t) dt + K_d \cdot \frac{d\varepsilon(t)}{dt} \quad (3.3)$$

Pour contrôler un processus industriel, on utilise en général une version numérique du régulateur PID qui reçoit en entrée la séquence des échantillonnées $c(k)$ calculée par :

$$c(k) = K_p \cdot \varepsilon(k) + k_i \cdot T_s \cdot \sum_{i=1}^n \varepsilon(i) + \frac{K_s}{T_e} \Delta \varepsilon(k) \quad (3.4)$$

Où T_e désigne la période d'échantillonnage, $\sum_{i=0}^n \varepsilon(i)$ est l'approximation de l'intégrale de l'erreur et $\Delta \varepsilon(k) = \varepsilon(k) - \varepsilon(k - 1)$ l'approximation de sa dérivée.

III.2.2. Réglage statique des gains d'un PID

Le problème du réglage des gains pondérant les différents termes du régulateur PID est un problème assez complexe abondamment traité dans la littérature.

La méthode la plus utilisée est celle de Ziegler-Nichols, car elle ne nécessite pas de connaître un modèle du processus commandé. Durant la phase de calibration, on remplace le PID par un simple amplificateur à gain réglable, comme le montre la figure (III.1). On augmente progressivement la valeur du gain afin d'amener le système en régime d'auto-oscillation (phénomène de pompage). La plus petite valeur du gain permettant d'obtenir un régime oscillant est notée K_p . On mesure alors la période des oscillations, notée T_u .

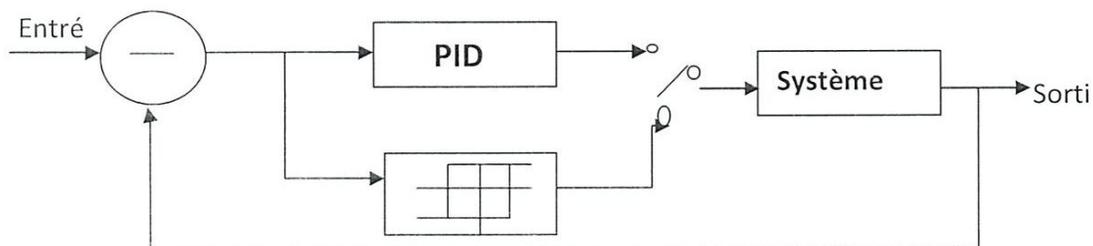


Fig.III.1. Réglage par Ziegler-Nichols des gains d'un PID

Les valeurs du gain proportionnel K_p et des constantes de temps intégrale T_i et dérivée T_p sont calculées par les formules suivantes :

$$K_p = 0.6K_u, T_i = 0.5T_u, T_d = 0.125T_u \quad (3.5)$$

Dans laquelle K_u est le gain et T_u la constante de temps qui ont été estimés lors de la phase de calibration.

III.2.3. Indices de performance d'un PID

Afin de définir la qualité de la régulation, on se base en général sur l'analyse de la réponse indicielle de l'ensemble régulateur PID plus système. La figure (III.2) montre l'allure générale d'une réponse à l'échelon :

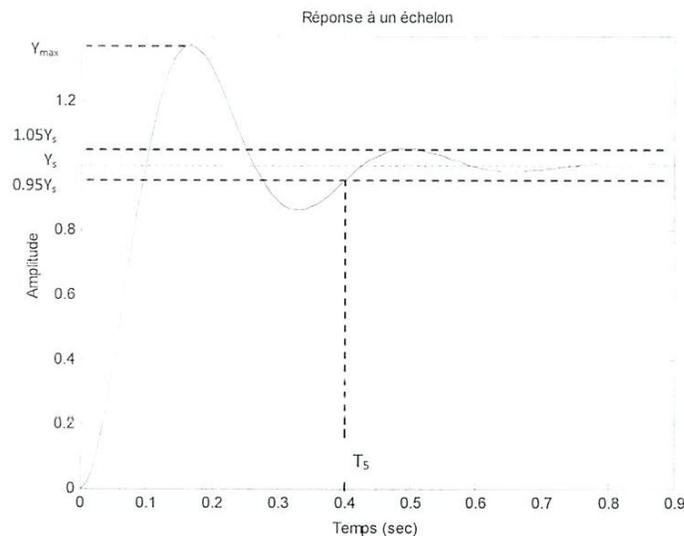


Fig III.2. Réponse à l'échelon d'un ensemble PID plus système

Différents indices de performance peuvent être évalués à partir de cette réponse temporelle. De façon générale, on cherche à quantifier la différence entre la réponse réelle du système asservi et une réponse idéale qui serait un échelon. Les indices couramment utilisés sont définis de la façon suivante :

- Y_{os} , *pourcentage de dépassement*. Avant de se stabiliser, la sortie du système passe par un régime transitoire oscillant de part et d'autre de la valeur finale. On définit le pourcentage de dépassement par :

$$Y_{os} = \frac{S_{max} - S(\infty)}{S(\infty)} \times 100\%, \quad (3.6)$$

Ou $S(\infty)$ est la valeur finale de la sortie et S_{max} est sa valeur maximale, qui est atteinte en général lors de la première oscillation.

$-T_5$, temps de stabilisation à 5%. Le temps de stabilisation à 5% de la valeur finale est une estimation de la durée du régime transitoire. On considère que le système atteint son régime permanent lorsque sa sortie est égale à la valeur finale à 5% près.

Cette durée est définie par :

$$T_5 = \min\{t_0 > 0 ; \forall t \geq t_0, s(\infty) \cdot (1 - 5\%) \leq s(t) \leq s(\infty) \cdot (1 + 5\%)\} \quad (3.7)$$

-Intégrales faisant intervenir l'erreur. Pour évaluer la différence existant entre la réponse réelle et une réponse idéale de type échelon, on peut calculer l'intégrale d'un terme positif faisant intervenir l'erreur. Un indice calculé de cette façon prend une valeur d'autant plus élevée que la réponse réelle est éloignée de la réponse idéale.

En pratique, l'intégrale est calculée sur un intervalle $[0, T]$ suffisamment étendu pour contenir tout le régime transitoire. L'intégrale de la valeur absolue de l'erreur $\varepsilon(t)$ donnée par :

$$IAE = \int_0^T |\varepsilon(t)| \cdot dt \quad (3.8)$$

On utilise également l'intégrale de l'erreur quadratique, définie par :

$$ISE = \int_0^T \varepsilon^2(t) \cdot dt \quad (3.9)$$

Pour pénaliser les systèmes dont le régime transitoire dure trop longtemps, on utilise également l'intégrale du produit de l'erreur par le temps, donnée par :

$$ITAE = \int_0^T t \cdot |\varepsilon(t)| \cdot dt \quad (3.10)$$

III.3. Réglage dynamique des gain d'un PID

Récemment, plusieurs auteurs ont décrit des méthodes de réglage dynamique des gains d'un PID par un contrôleur flou [8, 9,10].

On obtient ainsi un PID qualifié d'adaptatif, puisque ses paramètres de fonctionnement dépendent de l'état du système. Le PID adaptatif réalisé selon ce principe n'est plus un régulateur linéaire.

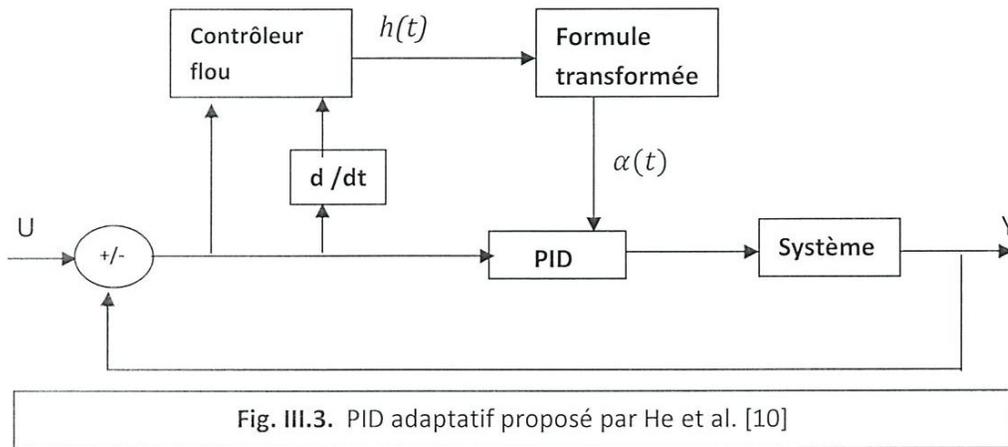
Dans la plupart de ces études, le contrôleur flou utilisé pour piloter le PID est défini par les auteurs à partir d'une série d'expériences. Pour tenter d'améliorer le réglage dynamique des gains, Wang et al. [8] utilisent un contrôleur flou dans lequel les règles floues ont été optimisées par un algorithme génétique standard. Alander et al. [9] utilisent également un contrôleur flou optimisé par un algorithme génétique à partir de critères définis dans le domaine fréquentiel.

He propose une structure de PID adaptatif dans lequel le réglage dynamique est superposé à un réglage statique réalisé par la méthode de Ziegler-Nichols. Les trois gains du PID sont donnés par les formules :

$$K_p = 1.2 \cdot \alpha \cdot K_u, \quad T_i = \frac{0.75}{1+\alpha} \cdot T_u, \quad T_d = 0.25 \cdot T_i, \quad (3.11)$$

Dans les quelles α est un coefficient réglé par le contrôleur flou. On constate aisément que les formules précédentes correspondent au réglage de Ziegler-Nichols lorsque α vaut 1/2 .

La structure du PID adaptatif proposé par He et al. [10] est représentée sur la figure (III.3).



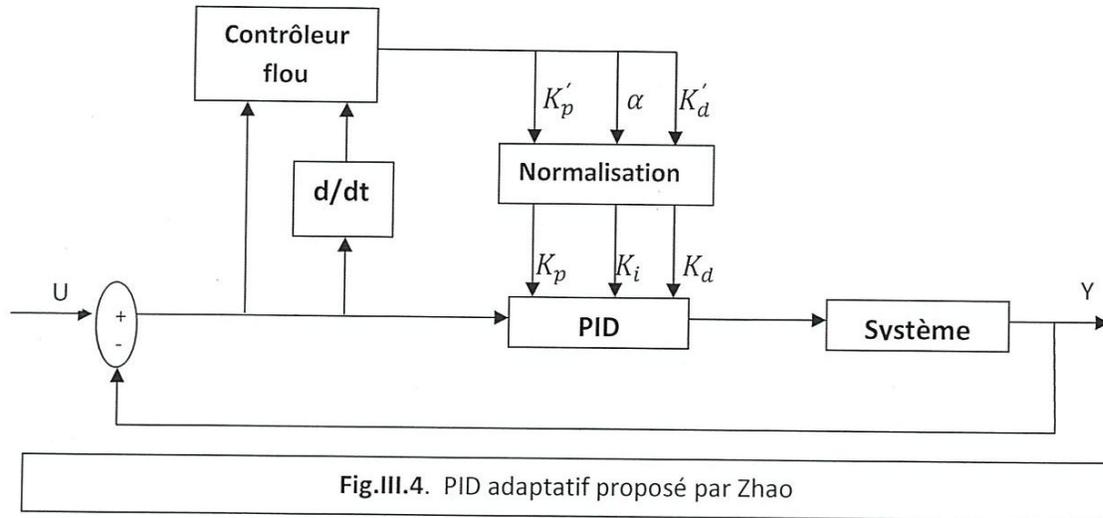
Le contrôleur flou utilisé par He est défini à partir d'une série d'expériences, et non par un procédé d'optimisation. Les entrées sont fuzzifiées en utilisant sept termes linguistiques dont les fonctions d'appartenance sont des Gaussiennes et les règles floues sont fixes.

III.4. Adaptation floue des paramètres d'un PID

Dans ce travail nous nous intéressons au réglage flou des gains du PID. Plus précisément au travail de Zhao [11] qui a proposé un PID adaptatif dans lequel un contrôleur flou ajuste dynamiquement les trois gains. La structure décrite par Zhao est représentée sur la figure (III.4).

Les gains K_p, K_i et K_d sont calculés à partir de trois valeurs intermédiaires fournies par le contrôleur flou, en utilisant les expressions suivant :

$$K_p = (K_{p,max} - K_{p,min}) \cdot K_p' + K_{p,min} \quad (3.12)$$



$$K_d = (K_{d,max} - K_{d,min}) \cdot K'_d + K_{d,min} \quad (3.13)$$

$$K_i = (k'_p / (\alpha \cdot K'_d)) \quad (3.14)$$

Où K'_p et K'_d prennent leur valeurs dans l'intervalle $[0,1]$ et α permet de régler la constante de temps intégrale en fonction de la constante de temps dérivée. Les valeurs extrêmes des gains sont définies à partir des paramètres K_u et T_u des formules de Ziegler-Nichols, par les expressions :

$$K_{p,min} = 0.32 \cdot K_u, \quad K_{p,max} = 0.6 \cdot K_u \quad (3.15)$$

$$K_{d,min} = 0.08 \cdot K_u \cdot T_u, \quad K_{d,max} = 0.15 \cdot K_u \cdot T_u \quad (3.16)$$

Zhao utilise une version échantillonnée du régulateur PID, dont l'entrée est alimentée par une série d'échantillons $\varepsilon(t)$. Le contrôleur flou utilise également les échantillons $\varepsilon(K)$ ainsi que l'estimation de leur dérivée temporelle donnée par la différence $\Delta\varepsilon(K) = \varepsilon(K) - \varepsilon(K-1)$ entre deux échantillons successifs. Les deux entrées du contrôleur flou sont limitées à l'intervalle $[-1,1]$ puis qualifiées par deux variables linguistiques identiques comportant sept termes. Les fonctions d'appartenance des sept termes linguistiques (NG, NM, NP, EZ, PP, PM, PG) sont de type triangulaire.

Les sorties du contrôleur flou sont qualifiées par des variables linguistiques dont les fonctions d'appartenance sont particulières. Pour les sorties K'_p et K'_d , Zhao utilise uniquement deux termes linguistiques P (pour petit) et G (pour grand), dont les fonctions d'appartenance sont logarithmiques et définies par :

$$\mu_p(x) = \min(1, -\frac{1}{4} \log x) \quad (3.17)$$

$$\mu_G(x) = \min(1, -\frac{1}{4} \log(1-x)) \quad (3.18)$$

Les 49 règles floues utilisées par Zhao comportent toutes deux prémisses et trois conclusions, et sont exprimées sous la forme :

Règle i : Si $(\varepsilon(k) \text{ est } A_i) \text{ et } (\Delta\varepsilon(k) \text{ est } B_i) \text{ alors } (K'_p \text{ est } C_i), (K'_d \text{ est } D_i), (\alpha = \alpha_i)$
 $i = 1, \dots, 49$

Ces règles sont représentées sous forme de tableaux III.1, III.2 et III.3. Les deux tableaux III.1 et III.2 montrent les conclusions des règles floues pour K'_p et K'_d qui sont qualifiées par P et G , et le tableau III.3 présente les conclusions des règles pour la sortie α , qui peut prendre quatre valeurs.

Zhao utilise la méthode d'inférence de Larsen, et la méthode COA (centre de région), pour la défuzzification de K'_p et K'_d . La sortie α ne doit pas être défuzzifiée puisque ses termes linguistiques sont des ensembles non flous.

		$\Delta\varepsilon(k)$							
		K'_p	NG	NM	NP	EZ	PP	PM	PG
$\varepsilon(k)$	NG	G	G	G	G	G	G	G	G
	NM	P	G	G	G	G	G	G	P
	NP	P	P	G	G	G	P	P	P
	EZ	P	P	P	G	P	P	P	P
	PP	P	P	G	G	G	P	P	P
	PM	P	G	G	G	G	G	G	P
	PG	G	G	G	G	G	G	G	G

Tableau III.1 : Conclusions des règles floues pour K'_p

		$\Delta\varepsilon(k)$							
		K'_d	NG	NM	NP	EZ	PP	PM	PG
$\varepsilon(k)$	NG	P	P	P	P	P	P	P	P
	NM	P	G	P	P	P	G	G	G
	NP	G	G	G	P	G	G	G	G
	EZ	G	G	G	G	G	G	G	G
	PP	G	G	G	P	G	G	G	G
	PM	G	G	P	P	P	G	G	G
	PG	P	P	P	P	P	P	P	P

Tableau III.2 : Conclusions des règles floues pour K'_d

		$\Delta\varepsilon(k)$							
		α	<i>NG</i>	<i>NM</i>	<i>NP</i>	<i>EZ</i>	<i>PP</i>	<i>PM</i>	<i>PG</i>
$\varepsilon(k)$	<i>NG</i>	2	2	2	2	2	2	2	2
	<i>NM</i>	3	3	2	2	2	3	3	
	<i>NP</i>	4	3	3	2	3	3	4	
	<i>EZ</i>	5	4	3	3	3	4	5	
	<i>PP</i>	4	3	3	2	3	3	4	
	<i>PM</i>	3	3	2	2	2	3	3	
	<i>PG</i>	2	2	2	2	2	2	2	

Tableau III.3 : Conclusions des règles floues pour α

Dans le PID adaptatif décrit précédemment, les règles floues intervenant dans le contrôleur ont été entièrement définies en utilisant des connaissances a priori sur le fonctionnement d'un PID. On sait, par exemple, qu'au début du régime transitoire le contrôleur doit réagir au plus vite à la modification de l'entrée. On doit donc disposer de gains proportionnel et intégral important, mais d'un gain dérivé assez faible. Cette connaissance est formalisée par la règle floue :

Si ($\varepsilon(k)$ est *PG*) et ($\Delta\varepsilon(k)$ est *PG*) alors (K'_p est *G*), (K'_d est *P*), ($\alpha = 2$)

III.5. Résultats

Dans ce travail nous testons le système flou d'adaptation des paramètres d'un PID sur le système de troisième ordre suivant :

$$G(p) = \frac{4.228}{(p+05)(p^2+1.64p+8.456)} \quad (3.19)$$

Ce système a été utilisé dans le travail original de Zhao [7] et dans [11].

Dans notre travail nous utilisons des fonctions d'appartenance triangulaire pour les entrées (erreur et dérivé de l'erreur) (figure III.5).

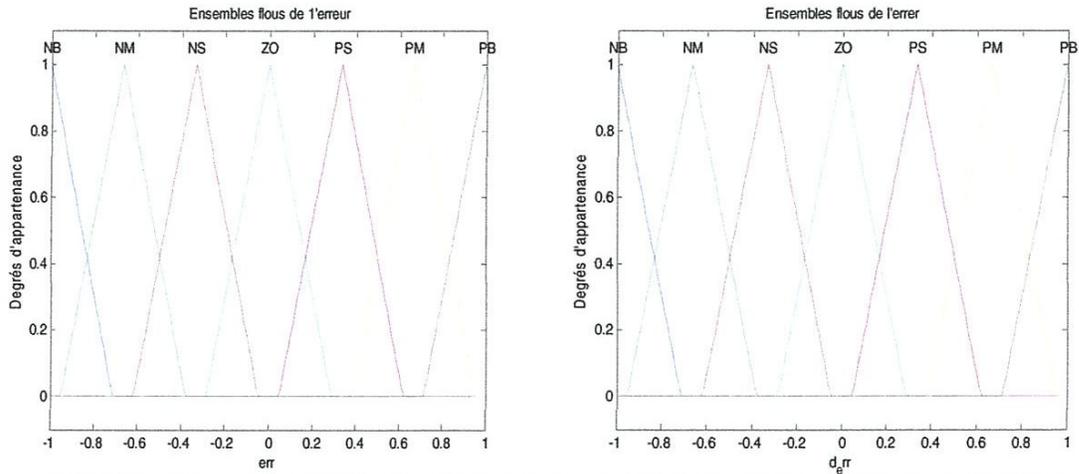


Fig III.5. Fonction d'appartenance de l'erreur et $d_{e,rr}$

Pour les sorties du système flou nous utilisons deux variables linguistiques P (pour petit) et G (pour grand) pour les sorties K'_p et K'_d . Les fonctions d'appartenance correspondantes sont données sur la figure (III.6). Pour la sortie α nous utilisons quatre fonctions d'appartenance gaussienne (figure III.7)

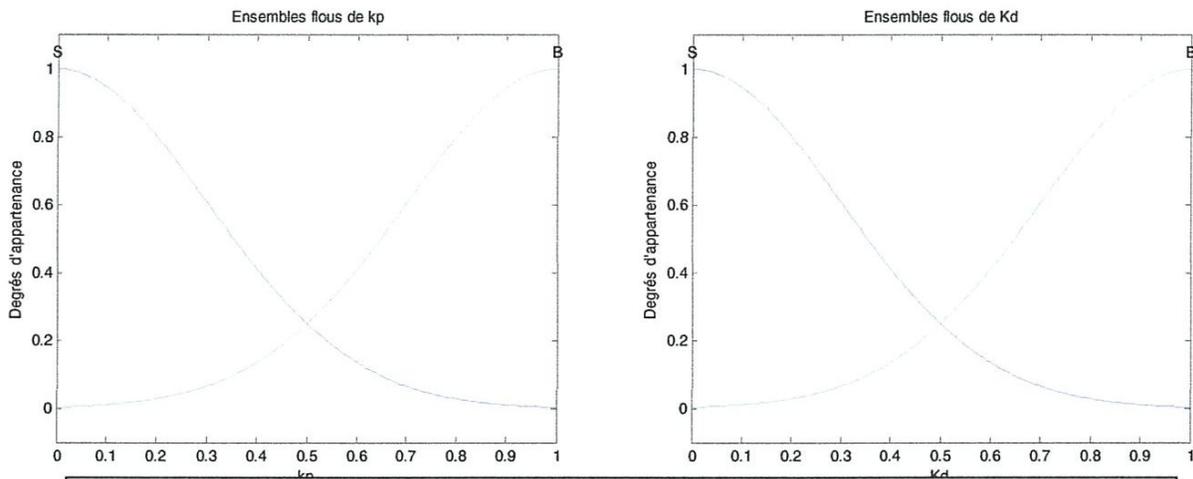


Fig III.6. Fonction d'appartenance K_p et K_d

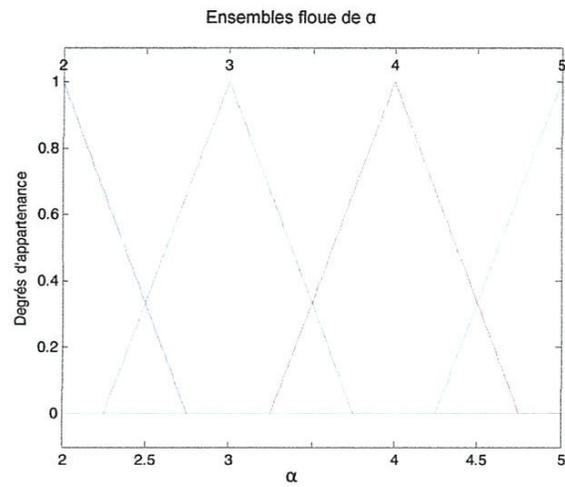


Fig III.7. Fonction d'appartenance α

Les figures (III.8, III.9, III.10) représentent la surface des règles correspondante aux trois sorties : K_p , K_d et α respectivement.

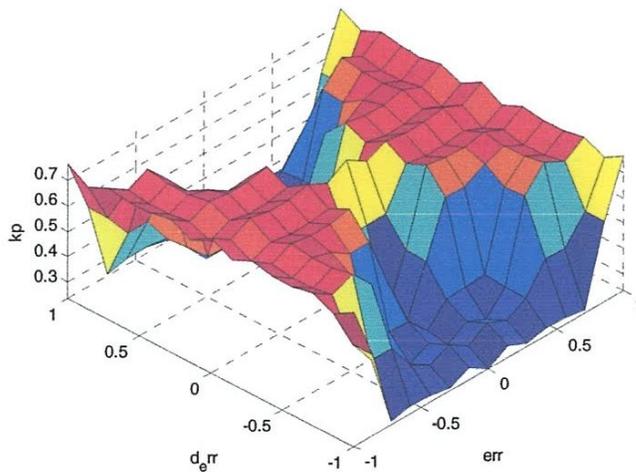


Fig III.8. Surface des règles de K_p

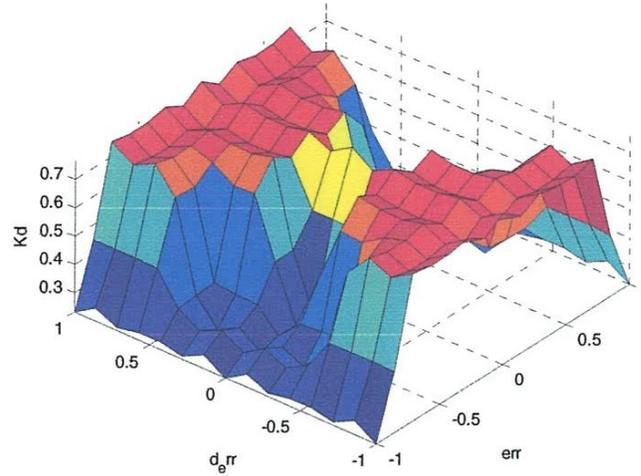


Fig III.9. Surface des règles K_d

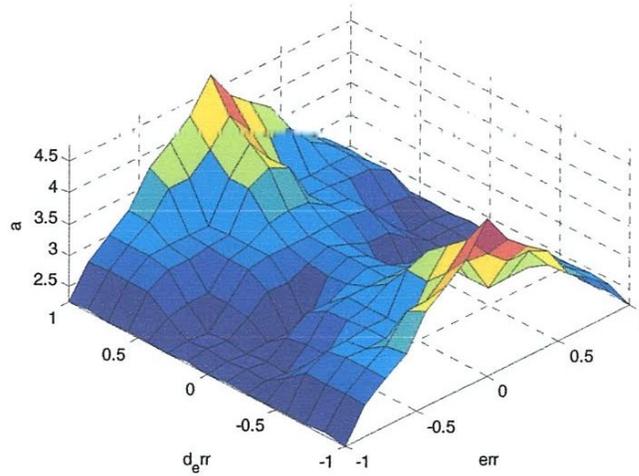


Fig III.10. Surface des règles α

Après simulation, l'évolution des paramètres dans la sortie du système flou (K_p , K_d et α) est représentée sur la figure (III.11).

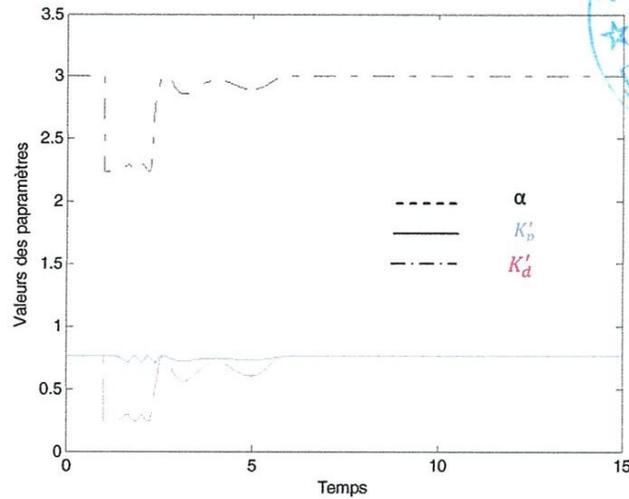


Fig III.11. Evolution des paramètres

La figure (III.12) illustre l'évolution des paramètres après normalisation et qui constituent les gains du PID.

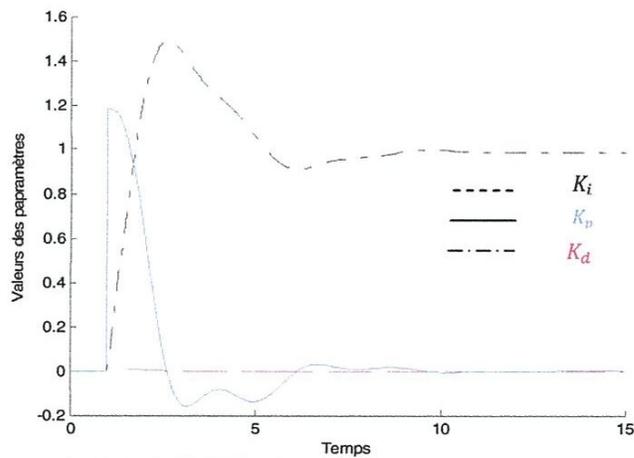


Fig III.12. Evolution des gains du PID

La figure (III.13) représente la réponse du système proposé et celui avec des gains fixe basé sur la méthode de Ziegler-Nichols. Nous pouvons constater que l'utilisation du système flou pour l'adaptation des gains du PID a accéléré le temps de stabilisation. Néanmoins, nous constatons également que le dépassement est presque le même.

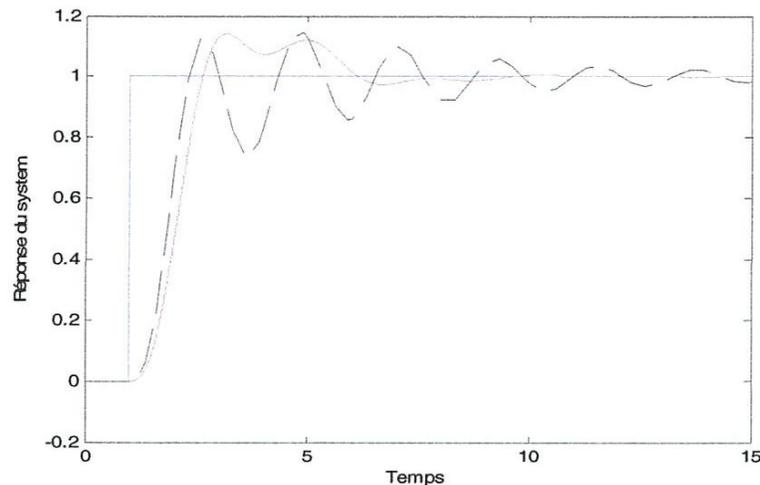


Fig III.13. Réponse du système

III.6. Conclusion

Nous avons évoqué les contrôleurs PID qui sont les contrôleurs les plus connus et les plus utilisés dans les processus de contrôle industriel en raison de leurs grandes performances dans le domaine industriel. Plus précisément, nous avons évoqué le réglage spécification de leurs trois paramètres : gain proportionnel, intégrale et dérivée en utilisant un système d'inférence flou. Les résultats de simulation ont montrons de grandes performances notamment en temps de réponse.