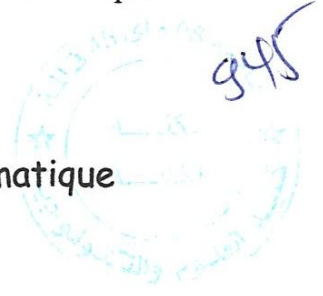


République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université 8 Mai 1945 – Guelma
Faculté des Sciences et de la Technologie
Département de Génie Electrotechnique et Automatique



Domaine: Sciences et Technologie
Filière: Automatique et informatique industriel
Spécialité: Commande et diagnostic des systèmes industriels

Projet de fin d'études
Pour l'obtention du diplôme de Master Académique

La commande par logique floue

Présenté par :

- **SALHI khayreddine**
- **AZZOUZ Sassi**

Sous la direction de :
Mr. BOUDJEHEM BADREDDINE

Jun 2017



REMERCIEMENTS

17/3467

*Avant tout on remercie le dieu tout puissant pour
Sa générosité de nous donner la santé, la volonté et
La patience tout au long de la vie universitaire.
Ensuite, nous adressons nos sincères remerciements
à nos enseignants notamment à notre promoteur
Boudjehem Badreddine. Qui franchement nous a
Donnée son maximum en vue d'élaborer un très bon
Mémoire ; sans oublier aussi ses conseils permanents,
Ses orientations efficaces et sa grande patience.*

On tient à exprimer notre profonde gratitude aux

Membres de jury.

*Nous remercions avec sincérité toutes les personnes
Qui ont nous contribué de près ou de loin à réussir ce
Travail.*

Nos vives salutations.

Dédicace

Avant tous, je remercie dieu le tout puissant de m'avoir donné le courage et la patience pour réaliser ce travail malgré toutes les difficultés rencontrées.

Nous dédions ce modeste travail :

A nos très chers parents, que dieu les garde et les protège pour leurs soutien moral et

A Nos frères, Nos sœurs et

Au

bilel, ali, zaki, samir, hicham, haroune, ALOULO, mohaned, yasmine,

Mohamed, tarek, marwan, soufi, houssam, lamine , riyad, ramzi,

Asma, rima, nina , zayneb, saaid , yazid, Zalarwa,

Zakaria, Salih , Cherif , Gouta , Housseem, Broken, Nawri,

, Mira , Wassimetc et a tout les supporteurs de f.c

Barcelona et A tous les amis (es) d'études surtout ceux d'automatique CDSI

Promotion 2017

Aux habitants de notre Wilaya Guelma

Sassi

Et

Khayreddine

Introduction générale	1
Chapitre I : Régulateurs Classiques	
I.1. Introduction	3
I.1.1 Nécessité de Correction dans les Systèmes Asservis	3
1. Stabilité	3
2. Précision	3
I.1.2. Différence Entre Asservir et Réguler	5
I.2. Organisation d'un Système en Boucle Fermée	5
I.3 Réponse d'un Système Asservi.....	5
a) La précision	6
b) La rapidité	6
c) La stabilité	7
I.4 Qualités Demandées à un Système	8
o Stabilité	8
o Poursuite.....	8
o Régulation	8
o Commande	9
I.5 Configurations de Correction	9
I.6 Correcteurs de Base	10
I.6.1 Correcteur à Action Proportionnelle (P)	10
I.6.1.1 Principe	10
I.6.1.2 Effet.....	11
I.6.2 Correcteur à Action Intégrale (I)	12
I.6.2.1 Principe	12
I.6.2.2 Effet.....	13
I.6.3 Correcteur à Actions Proportionnelle et Intégrale (PI)	13

I.6.3.1 Principe	14
I.6.3.2 Effet.....	15
I.6.4 Correcteur à Action Dérivée (D)	15
I.6.4.1 Principe	15
I.6.4.2 Effet.....	17
I.6.5 Correcteur à Actions Proportionnelle et Dérivée (PD)	17
I.6.5.1 Principe	17
I.6.5.2 Effet.....	18
I.6.6 Correcteur à Actions Proportionnelle, Intégrale et Dérivée (PID).....	18
I.6.6.1 Principe	19
I.6.6.2 Effet.....	20
I.7 Conclusion.....	22
Chapitre II : La logique floue	
II.Introduction	23
II.1 Historique de la logique floue	24
II.2 Principe de la Logique Floue.....	26
II.3 Sous-ensembles flous	28
II.3.1 Définitions.....	28
Définition 1.1 (Sous-ensemble flou).....	29
Définition 1.2 (Support)	30
Définition 1.3 (Point de croisement).....	30
Définition 1.4 (Noyau)	30
Définition 1.5 (Hauteur)	30
II.4 Différentes formes des fonctions d'appartenance.....	31
a) fonction d'appartenance triangulaire.....	31
b) fonction d'appartenance trapézoïdale.....	32
c) fonction d'appartenance gaussienne.....	32

II.4.1 Opérations sur les sous-ensembles flous	32
Définition 1.6 (Egalité)	32
Définition 1.7 (Inclusion)	33
Définition 1.8 (Intersection)	33
Définition 1.9 (Union).....	34
Définition 1.10 (Complément)	34
II.4.2 Normes et conormes triangulaires.....	35
Définition 1.11 (Norme triangulaire, t-norme)	35
II.4.3 Relations floues.....	35
II.5 Raisonnement en logique floue.....	36
II.5.1 Variables linguistiques.....	36
II.5.2 Propositions floues.....	38
Définition 1.12 (Propositions floues élémentaires).....	38
Définition 1.13 (Propositions floues générales).....	39
II.5.3 Implications floues.....	40
Définition 1.14 (Implications floues).....	41
II.5.4 Inférence floue	42
II.6 Conclusion.....	42
 Chapitre III : La Commande floue	
III. Introduction.....	43
III.1 Fuzzification	44
III.2 Règles floues	46
III.3 Inférences floues.....	46
III.3.1 Inférence floue de Mamdani.....	47
III.3.2 Inférence floue de Sugeno.....	48
III.4 Défuzzification	49
III.4.1 Défuzzification par centre de gravité.....	50
III.4.2 Défuzzification par centre maximum.....	51
III.4.3 Défuzzification par valeur maximum	51

III.5 Avantages et Inconvénients de Réglage par Logique Floue	51
III.5.1 Avantages	51
III.5.2 Inconvénients	51
III.6 Conclusion.....	52
Chapitre IV : Application	
IV.1 Introduction.....	53
IV.2 Régulation avec un Correcteur PID.....	54
IV.3 Régulation par logique floue.....	56
IV.3.1 Régulation à une seule entrée.....	56
IV.3.2 Régulateur flou à deux entrées.....	60
IV.4 Autre exemple explicatif.....	65
IV.5 Conclusion.....	68
Conclusion générale.....	69
Bibliographié.....	70

LISTE DES FIGURES

Figure I.1 : Quantification de la stabilité et de la précision sur le diagramme de Bode.....	4
Figure I.2 : Système en boucle fermée.....	5
Figure I.3 : Réponse d'un système asservi.....	6
Figure I.4 : Réponse d'un système asservi.....	7
Figure I.5 : Réponse d'un système asservi.....	7
Figure I.6 : Réponse d'un système asservi.....	8
Figure I.7 : Processus commandé.....	9
Figure I.8 : Correction en cascade ou série.....	10
Figure I.9 : Correction P.....	11
Figure I.10 : Entrée et sortie du correcteur P.....	11
Figure I.11 : Correction I.....	13
Figure I.12 : Entrée et sortie du correcteur I.....	13
Figure I.13 : Correction PI.....	14
Figure I.14 : Entrée et sortie du correcteur PI.....	14
Figure I.15 : Correction à action dérivée pure (D).....	15
Figure I.16 : Entrée et sortie du correcteur à action dérivée (D).....	15
Figure I.17 : Correction PD.....	18
Figure I.18 : Entrée et sortie du correcteur PD.....	18
Figure I.19 : Correction à actions proportionnelle, intégrale et dérivée (PID).....	19
Figure I.20 : Entrée et sortie du correcteur à actions (PID).....	20
Figure II.1 : classification des températures d'une pièce en deux ensembles.....	26
Figure II.2 : Classification selon la logique classique.....	27
Figure II.3 : Classification selon la logique floue.....	27
Figure II.4 : Différentes formes de la fonction d'appartenance.....	32
Figure II.5 : Intersection des fonctions d'appartenances.....	33
Figure II.6 : Union des fonctions d'appartenances.....	34
Figure II.7 : Fonction d'appartenances.....	34

Figure II.8 : t-normes et t-conormes les plus utilisées.....	36
Figure II.9 : Représentation des variables linguistiques.....	37
Figure II.10 : Variable linguistique.....	38
Figure II.11 : Valeur de vérité de la proposition floue p : la température est tiède.....	39
Figure III.1 : structure générale d'un contrôleur floue	44
Figure III.2 : Univers du discours partitionné par les termes linguistique définis dans le tableau III.1.....	45
Figure III.3: Désignations standard et fonctions d'appartenance.....	46
Figure III.4 : Exemple d'inférence max-min.....	48
Figure III.5 : Exemple d'inférence max-prod.....	49
Figure III.6 : Exemple d'inférence de Mamdani.....	50
Figure IV.1 : Une serre agricole.....	53
Figure IV.2 : Mise en oscillation juste entretenus d'un système.....	54
Figure IV.3 : Régulation avec un correcteur PID.....	55
Figure IV.4: Réponse du système avec un correcteur PID.....	56
Figure IV.5 : Schéma Simulink d'un correcteur floue a une seule entrée.....	56
Figure IV.6 : Interface de la logique floue sous Matlab.....	57
Figure IV.7: Fonctions d'appartenance de l'entrée (erreur) et de la sortie (commande).....	57
Figure IV.8 : Réponse du système avec un correcteur floue a une seule entrée	59
Figure IV.9: Réponses du système avec un correcteur PID et floue	59
Figure IV.10 : Interface de la logique floue sous Matlab.....	60
Figure IV.11: Fonctions d'appartenance de l'entrée (erreur).....	61
Figure IV.12: Fonctions d'appartenance de l'entrée (derreur).....	61
Figure IV.13: Fonctions d'appartenance de sortie (commande).....	61
Figure IV.14 : Schéma Simulink d'un correcteur floue a deux entrées	62
Figure IV.15 : Commande PID pour un retard= 1.5s.....	63
Figure IV.16 : Commande floue pour un retard= 1.5s.....	63

Figure IV.17 : Commande PID pour un retard= 2 s.....	64
Figure IV.18 : Commande floue pour un retard= 2s.....	64
Figure IV.19 : Commande PID pour un retard= 3s.....	65
Figure IV.20 : Schéma Simulink d'un correcteur floue et PID.....	66
Figure IV.21: Fonctions d'appartenance de l'entrée (erreur).....	67
Figure IV.22: Fonctions d'appartenance de sortie (commande).....	67
Figure IV.23: Réponses du système avec un correcteur PID et floue	68

LISTE DES TABLEAUX

Tableau I.1 : Effets des correcteurs P, I, et D sur les régimes statique et dynamique du système en boucle fermée.....	20
Tableau II.1 : Implications floues les plus utilisées.....	41
Tableau III.1 : Exemple de variable linguistique.....	45
Tableau IV.1 : Réglages préconisés par Ziegler et Nichols.	55
Tableau IV.2 : les règles floues.....	62

Résumé

Les méthodes conventionnelles de réglage sont basées sur une modélisation adéquate du système à régler et un traitement analytique à l'aide de la fonction de transfert ou d'équations d'état. Malheureusement, celles-ci ne sont pas toujours disponibles,

Les commandes à base de logique floue ont connu un succès croissant depuis la fin du siècle dernier, notamment dans le domaine du génie électrique. Celles-ci apportent en effet une amélioration significative des performances par rapport à des commandes linéaires plus classiques. Cependant, les paramètres de ce type de commande sont nombreux et délicats à régler. Le fil conducteur de ces travaux consiste alors à proposer des méthodologies de réglage simples pour des commandes. Dans ce travail, nous avons présenté la commande des systèmes par la logique floue. Les lois de commande classique du type PI ou PID donnent des bons résultats dans le cas des systèmes linéaires à paramètres constants. Pour des systèmes non linéaires où ayant des paramètres non constants, ces lois de commande classique peuvent être insuffisantes car elles sont non robustes surtout lorsque les exigences sur la précision et autres caractéristiques dynamiques du système sont strictes, on doit faire appel à des lois de commande insensibles aux variations de paramètres aux perturbations et aux non linéarités.

Le but de notre travail est d'étudier en simulation l'application d'une commande de régulation de la température comme exemple simple, puis de faire une comparaison avec la commande analogique PID du même procédé à l'aide du logiciel Simulink (Matlab).

L'industrie évolue d'une production artisanale-traditionnelle à des procédés de plus en plus automatisés. Deux facteurs encouragent cette évolution : d'une part la tendance de toute industrie est de devenir de plus en plus rentable et compétitive afin d'assurer sa permanence dans le système économique actuel ; d'autre part, les progrès technologiques et scientifiques mettent à disposition des outils mathématiques et informatiques de plus en plus performants, à l'aide desquels les caractéristiques et besoins particuliers d'un grand spectre de procédés différents (et de produits) peuvent être maîtrisés.

La rentabilité du système de fabrication suppose l'amélioration sans cesse des conditions d'exploitation et de maîtrise des procédés. Ces procédés constituent l'ensemble de moyens technologiques (et de leurs règles de conduite) qui permettent de conférer, ou d'inhiber, des propriétés à un produit.

En conséquence, la plus grande partie des travaux sur l'optimisation est menée soit en absence de modèles mathématiques, soit avec un modèle mathématique (simple) empirique. Dans ces cas, les résultats obtenus restent modestes aux prix de grands efforts expérimentaux (dont les données obtenues sont fréquemment sous-exploitées).

D'autre part, les résultats de l'optimisation dépendent de la 'bonne' construction du problème d'optimisation et du choix 'adéquat' de la méthode de résolution de ce problème. 'Bien' construire le problème d'optimisation signifie établir clairement *les variables de commande*, *les critères* (les objectifs de l'optimisation) et *les contraintes* afin de limiter l'espace des solutions admissibles. Alors que la méthode 'adéquate' est celle qui est la plus adaptée aux caractéristiques du problème d'optimisation, dans certains cas il est convenable d'utiliser une combinaison des méthodes pour obtenir les meilleurs résultats.

La complexité du modèle utilisé, même si celle-ci suppose le déroulement de grands efforts de calcul, ne pose plus de problème pour les optima calculés hors ligne grâce à l'ensemble des méthodes numériques et des outils informatiques disponibles actuellement. Une bonne stratégie d'optimisation consiste alors à utiliser le modèle mathématique le plus fiable, à poser convenablement le problème d'optimisation et, enfin, à choisir la méthode de solution la plus adaptée aux besoins.

En effet les systèmes devenant de plus en plus complexes, 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. Dans ce cas, les régulateurs conventionnels sont difficilement utilisables.

Afin de régler dynamiquement les paramètres d'un système, nous utilisons un contrôleur flou. Ce principe de régulation, basé sur des concepts relativement simples, 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.

La théorie de la logique floue a pour principal intérêt de prendre en compte l'analyse qualitative d'un expert. Les commandes floues dérivées de ce principe n'utilisent alors aucun modèle explicite du système ; elles sont en effet basées sur l'intégration de la stratégie de commande d'un expert. On évite par conséquent les problèmes d'une modélisation complexe d'un système. Les commandes floues ont montré par le succès de leurs utilisations dans des applications industrielles leur utilité dans le contrôle des systèmes.

Néanmoins, la commande floue n'est pas exempte de problèmes dont les plus cruciaux sont le réglage des différents paramètres et la difficulté de dominer le comportement transitoire du système commandé.

Nos travaux ont pour objectif de contribuer à résoudre ces deux problèmes, tout d'abord par l'analyse d'un système flou et ensuite par l'utilisation de la théorie des possibilités dérivée de la logique floue.

Chapitre I :

Régulateurs Classiques

I.1. Introduction :

I.1.1 Nécessité de Correction dans les Systèmes Asservis :

Pour satisfaire aux spécifications de stabilité et de précision, on est amené à formuler des conditions sur la Fonction de Transfert en Boucle Ouverte (FTBO) :

1. Stabilité :

Le degré de stabilité est défini par :

La marge de gain : la stabilité est d'autant meilleure que :

- le gain de la FTBO est plus faible,
- donc, que la bande passante en BO est plus faible.

La marge de phase : la stabilité est d'autant meilleure que :

- le déphasage de la FTBO est plus faible.

2. Précision :

Son étude se décompose en deux parties :

- Précision statique : l'annulation de l'erreur en régime permanent nécessite la présence, dans la FTBO, d'une ou plusieurs intégrations selon l'entrée canonique imposée.
- Précision dynamique : elle est d'autant meilleure que le gain de la FTBO est plus élevé, c'est-à-dire que la bande passante est plus large.

Pour simplifier à l'extrême, on peut retenir, en résumé que la précision et la stabilité sont quantifiées, dans le diagramme de Bode de la FTBO, de la manière suivante (Figure I.1) :

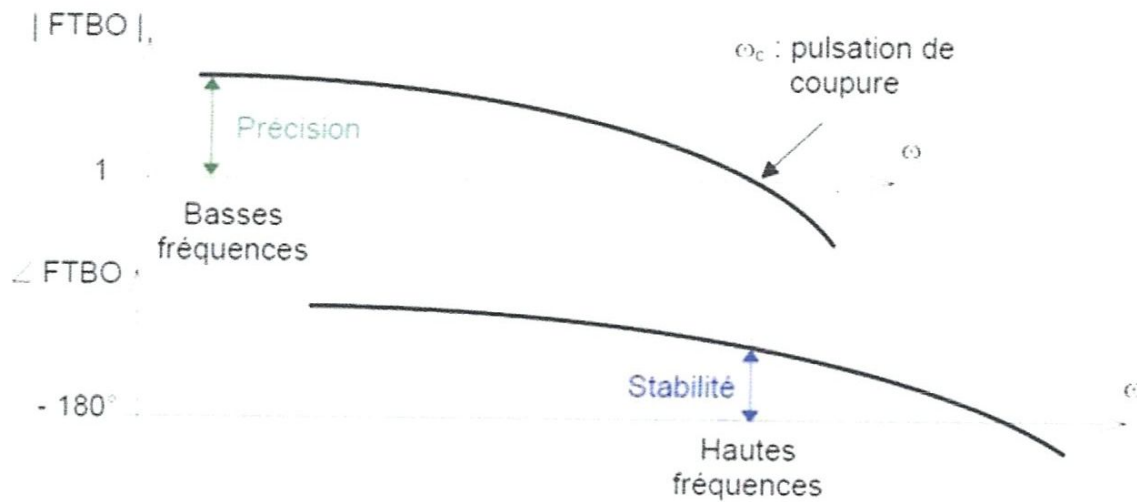


Figure I.1 : Quantification de la stabilité et de la précision sur le diagramme de Bode

Il devient clair que :

Pour améliorer la précision, il faut pouvoir augmenter le gain de la FTBO.

La stabilité diminue si ce même gain devient trop élevé.

Il semble donc difficile d'obtenir un système, à la fois, précis (grand gain) et stable (faible gain). Ce premier dilemme stabilité-précision impose donc l'emploi de systèmes compensateurs, correcteurs, ou encore régulateurs dont le rôle sera de relever le gain dans une certaine zone de fréquence et de la diminuer ailleurs.

Le correcteur ou régulateur va permettre de satisfaire les contraintes suivantes :

Trouver un compromis entre la stabilité et la précision,

- Si besoin, rendre stable en boucle fermée un système qui serait instable en boucle ouverte,
- Si besoin, et c'est en général le cas, introduire un intégrateur dans la boucle pour obtenir une erreur statique nulle ($\varepsilon = 0$).

Mais, il existe d'autres incompatibilités qui nécessitent également l'emploi de correcteurs :

Un bon asservissement doit être insensible aux perturbations et, en même temps, il doit répondre rapidement aux variations des diverses grandeurs d'entrée. Ces deux conditions sont incompatibles puisque :

La réponse rapide nécessite une large bande passante, tandis que l'insensibilité aux perturbations exige une bande étroite.

Le rôle des correcteurs, qui peuvent être électriques, mécaniques, ou hydrauliques, est donc de déformer le diagramme asymptotique ou la courbe de Nyquist pour leur donner des marges de gain et de phase capables d'assurer la stabilité tout en conservant aux basses fréquences un gain suffisamment grand pour que la précision soit bonne.

I.1.2. Différence Entre Asservir et Réguler :

Un système en boucle fermée peut remplir la fonction :

Asservissement : poursuite par la sortie d'une consigne variable dans le temps,

Régulation : la consigne est constante, le système compense les perturbations.

I.2. Organisation d'un Système en Boucle Fermée :

Dans un système en boucle fermée, on trouve les éléments suivants :

- Un capteur pour mesurer la sortie,
- Un comparateur qui élabore l'erreur entre la consigne et la mesure de la sortie, un correcteur qui élabore la commande en fonction du signal d'erreur,
- Un organe de commande qui module le signal d'entrée du système.

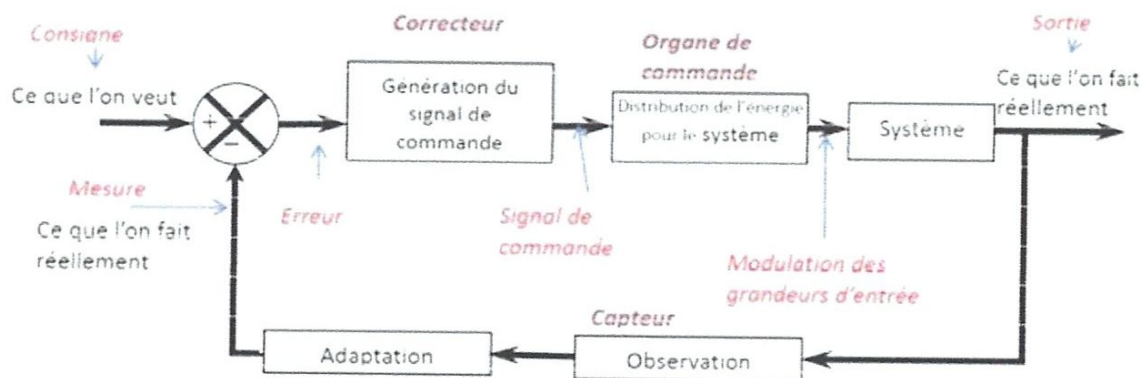


Figure I.2 : Système en boucle fermée

I.3 Réponse d'un Système Asservi :

Un système asservi est caractérisé par :

a) La précision

C'est la capacité du système à se rapprocher le plus possible de la valeur de consigne.

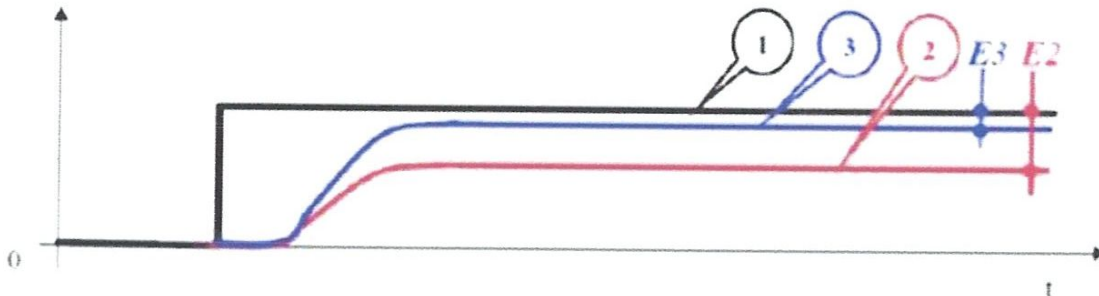


Figure I.3 : Réponse d'un système asservi

1 : consigne

2 : système peu précis.

3 : système précis.

E2 : erreur statique liée à la courbe 2.

E3 : erreur statique liée à la courbe 3.

L'erreur s'exprime en pourcentage de la valeur de consigne.

b) La rapidité

C'est la capacité du système à atteindre dans les meilleurs délais son régime stable. La rapidité d'un système est définie par son temps de réponse t_r (plus t_r est petit plus le système est dit rapide).

Dans l'exemple suivant, la courbe noire (1) représente la consigne et la bleue (2) représente la réponse du système. La valeur finale du système est nommée v_f .

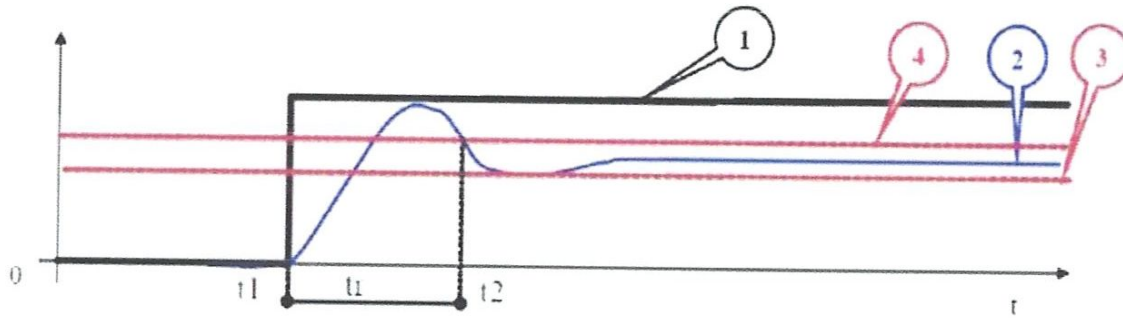


Figure I.4 : Réponse d'un système asservi

- 1 : consigne.
- 2 : valeur finale du système.
- 3 : droite à 95% de la valeur finale.
- 4 : droite à 105% de la valeur finale.

Pour déterminer le temps de réponse d'un système :

On trace une droite à 95% de la valeur finale v_f (3);

On trace ensuite une droite à 105% de la valeur finale v_f (4).

Le temps de réponse à 5 % correspond à la différence entre le temps t_2 (temps à partir duquel la courbe entre dans l'intervalle 95% /105% sans en sortir) et le temps t_1 (temps à partir duquel la consigne est active).

c) La stabilité

Pour une consigne constante la sortie doit tendre vers une constante.

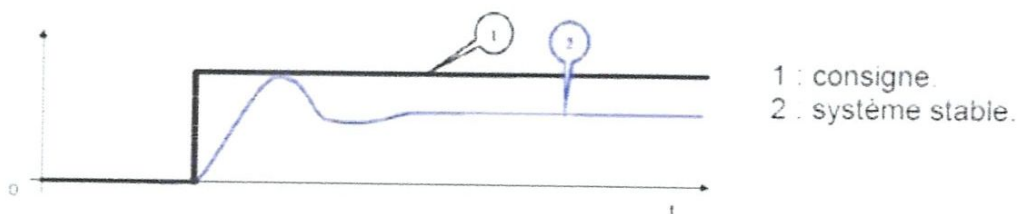


Figure I.5 : Réponse d'un système asservi

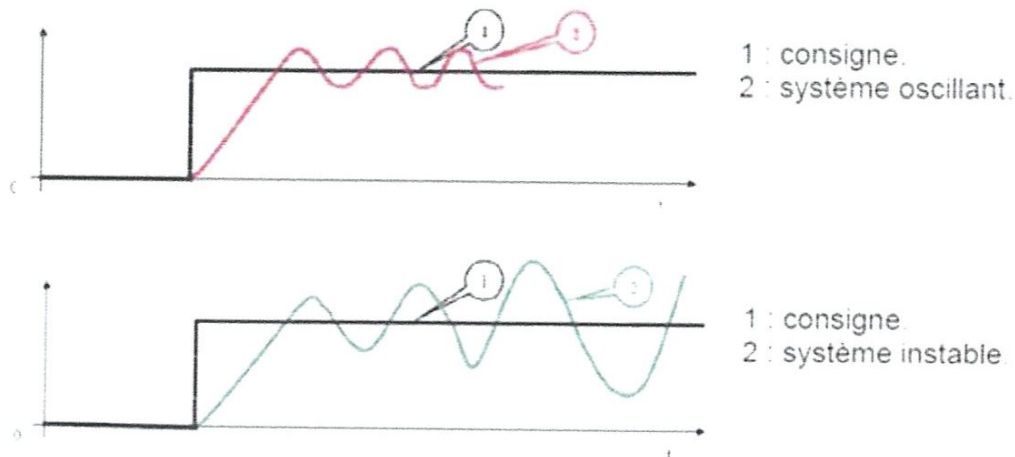


Figure I.6 : Réponse d'un système asservi

I.4 Qualités Demandées à un Système :

Stabilité

C'est la première qualité demandée à un système. Pour pouvoir en effectuer la correction, il faut absolument respecter cette condition. Cela implique que tous les pôles de la fonction de transfert soient à partie réelle négative. Dans le cas où le modèle du système considéré est instable, le correcteur aura la charge de « ramener » les pôles instables du côté stable (demi-plan complexe gauche) et donc de stabiliser le système.

Poursuite

Le problème de poursuite consiste à ce que la sortie du système « suive au mieux » son entrée (en général, la consigne). Le système doit être :

Précis : en régime permanent, la sortie doit atteindre « au plus près » le niveau demandé par la consigne;

Rapide : la sortie doit atteindre la valeur de consigne le plus rapidement possible. Le temps de réponse doit être le plus petit possible. La situation des pôles dans le plan complexe en est responsable.

Régulation

Le système est soumis à une (ou des) perturbation(s). Malgré celle-ci, il doit être :

Précis : la sortie doit atteindre « au plus près » le niveau de l'entrée même si une perturbation vient tenter de l'en écarter ;

Rapide : il doit effacer l'influence de la perturbation. La sortie doit « revenir » le plus vite possible au niveau souhaité.

Commande :

La commande appliquée au système doit être limitée (notamment en amplitude) afin d'éviter un mauvais fonctionnement : au mieux l'actionneur sature et la commande appliquée au système est maximale (dans ce cas, il y a plus de régulation), au pire on risque de détruire l'actionneur ou le système.

Ces qualités s'évaluent en général en appliquant au système une entrée en forme d'échelon. On verra dans la pratique que ces exigences de performances ne peuvent être satisfaites de manière idéale. On adoptera alors un compromis en fonction des priorités du cahier des charges.

I.5 Configurations de Correction :

En général, la dynamique d'un processus commandé peut être représentée par le schéma fonctionnel de la figure I.7.

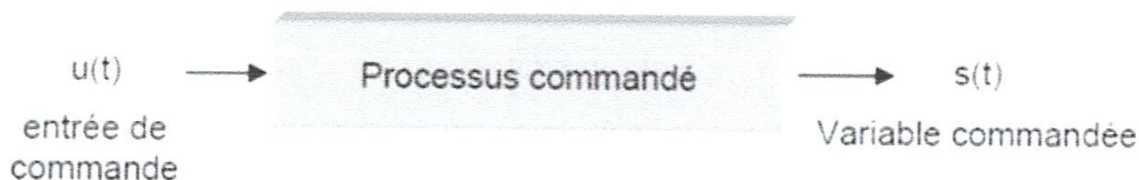


Figure I.7 : Processus commandé

L'objectif est que la variable commandée, représentée par la sortie $s(t)$, ait un comportement désiré sur un intervalle de temps donné. Il s'agit alors de déterminer le signal de commande $u(t)$ qui, dans cet intervalle, garantisse la sortie $s(t)$ désirée [2].

On peut réaliser la compensation en insérant, dans une chaîne, un correcteur directement en cascade avec les autres éléments (Figure I.8).

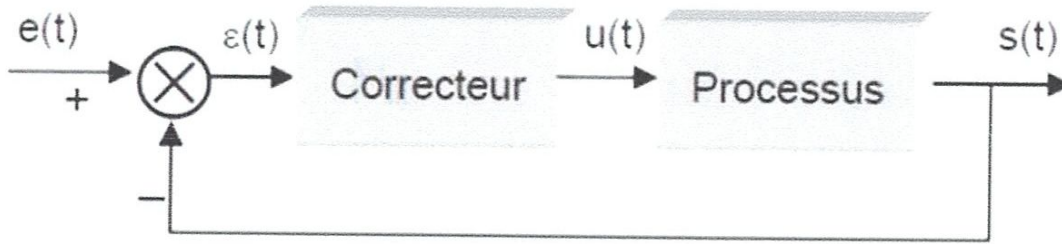


Figure I.8 : Correction en cascade ou série

I.6 Correcteurs de Base :

Un correcteur est un algorithme de calcul qui délivre un signal de commande à partir de la différence entre la consigne et la mesure.

Les correcteurs industriels les plus utilisés peuvent être classés, selon leurs actions de correction [1], de la manière suivante :

- Correcteur à action proportionnelle (P),
- Correcteur à action intégrale (I),
- Correcteur à actions proportionnelle et intégrale (PI),
- Correcteur à action dérivée (D),
- Correcteur à actions proportionnelle et dérivée (PD),
- Correcteur à actions proportionnelle, intégrale et dérivée (PID).

I.6.1 Correcteur à Action Proportionnelle (P) :

I.6.1.1 Principe :

Le correcteur à action proportionnelle, est le correcteur le plus simple, puisque défini par un simple gain K .

La relation entre la sortie du correcteur (ou signal de commande) $u(t)$ et le signal d'entrée du correcteur (signal d'erreur) $\varepsilon(t)$ est :

$$u(t) = K_p \cdot \varepsilon(t)$$

Sa fonction de transfert est donc :

$$C(p) = \frac{U(p)}{\varepsilon(p)} = K_p$$

Où K_p : gain proportionnel.

Quelques soient le mécanisme et la source d'énergie utilisés, le correcteur proportionnel est essentiellement un amplificateur à gain variable. Son schéma fonctionnel est celui de la figure I.9.

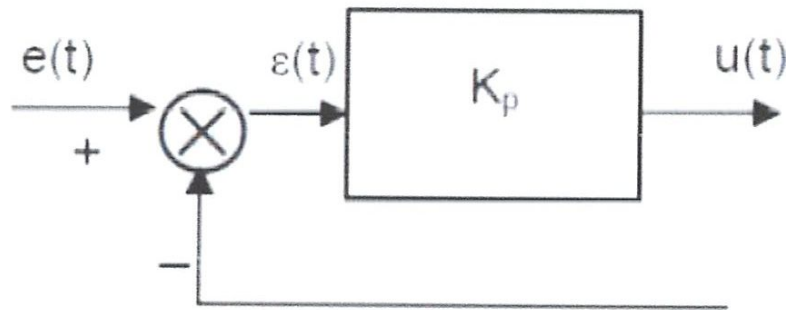


Figure I.9 : Correction P

La figure I.10 donne la réponse indicielle du correcteur P.



Figure I.10 : Entrée et sortie du correcteur P.

I.6.1.2 Effet :

L'action proportionnelle P crée un signal de commande $u(t)$ proportionnel au signal d'erreur $\varepsilon(t)$. Elle agit donc principalement sur le gain du système asservi et permet d'améliorer notablement la précision.

L'action proportionnelle :

- entraîne une augmentation du gain, d'où une diminution de l'erreur statique (Amélioration de la précision), mais
- augmente la bande passante du système, ce qui
- améliore la rapidité du système et,
- augmente l'instabilité du système.

Le correcteur proportionnel P n'est généralement pas utilisé seul. On verra que tout correcteur possède au moins l'action proportionnelle.

On constate que l'augmentation de K_p , entraîne :

- une amélioration de l'erreur statique,
- une décroissance du temps de montée,
- une faible amélioration du temps d'établissement,
- mais également une diminution de la marge de phase et une augmentation du dépassement (augmentation de l'instabilité du système).

I.6.2 Correcteur à Action Intégrale (I) :

I.6.2.1 Principe :

La relation entre la sortie $u(t)$ et le signal d'erreur $\varepsilon(t)$ est :

$$u(t) = K_i \int_0^t \varepsilon(\tau) d\tau$$

Ou encore :

$$\frac{du(t)}{dt} = K_i \cdot \varepsilon(t)$$

Sa fonction de transfert est donc :

$$C(p) = \frac{U(p)}{\varepsilon(p)} = \frac{K_i}{p} = \frac{1}{T_i p}$$

Où :

K_i : gain intégral.

T_i : constante de temps d'intégration.

Son schéma fonctionnel est celui de la figure I.11.

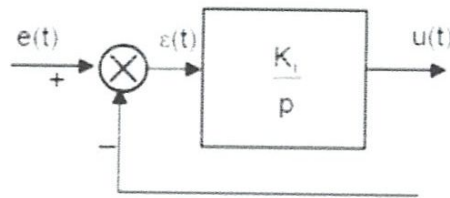


Figure I.11 : Correction I

La figure I.12 donne la réponse indicielle du correcteur Intégrale.

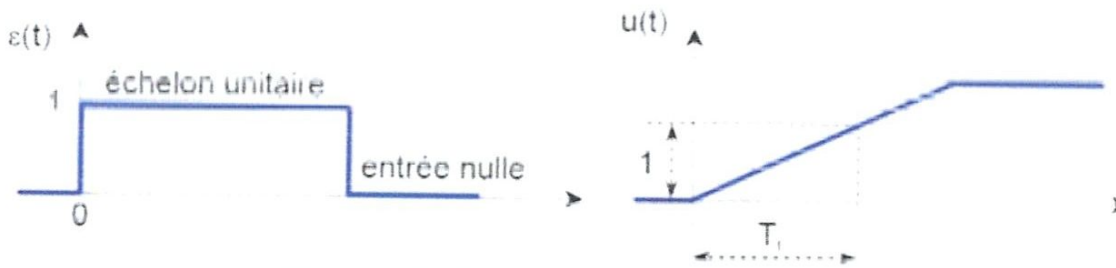


Figure I.12 : Entrée et sortie du correcteur I

I.6.2.2 Effet :

L'intérêt principal de ce correcteur est d'ajouter dans la chaîne de commande une intégration. Nous savons que la présence d'une intégration dans la FTBO augmente la classe du système et réduit ou annule, selon le type d'entrée, l'erreur statique du système.

L'action intégrale pure :

- Améliore la précision en réduisant ou annulant l'erreur statique, mais
- Introduit un déphasage de -90° qui risque de déstabiliser le système (diminution de la marge de phase).

Le correcteur à action exclusivement Intégrale n'est pratiquement jamais utilisé, en raison de sa lenteur et de son effet déstabilisant. Il est, en général, associé au correcteur Proportionnel.

I.6.3 Correcteur à Actions Proportionnelle et Intégrale (PI) :

I.6.3.1 Principe :

La relation entre la sortie $u(t)$ et le signal d'erreur $\varepsilon(t)$ est :

$$u(t) = K_p \cdot \varepsilon(t) + K_i \int_0^t \varepsilon(t) dt$$

Sa fonction de transfert est donc :

$$C(p) = \frac{U(p)}{\varepsilon(p)} = K_p + \frac{K_i}{p} = K_p \left(1 + \frac{K_i}{K_p p} \right) = K_p \left(1 + \frac{1}{T_n p} \right)$$

Où :

K_p : gain proportionnel.

K_i : gain intégral.

$T_i = \frac{1}{K_i}$: constante de temps d'intégration.

$T_n = \frac{K_p}{K_i}$: dosage de corrélation d'intégrale.

Son schéma fonctionnel est celui de la figure I.13.

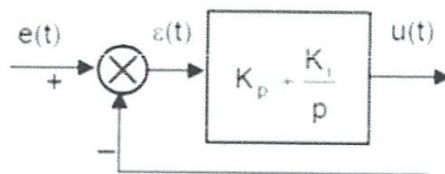


Figure I.13 : Correction PI

La figure I.14 donne la réponse indicielle du correcteur PI.

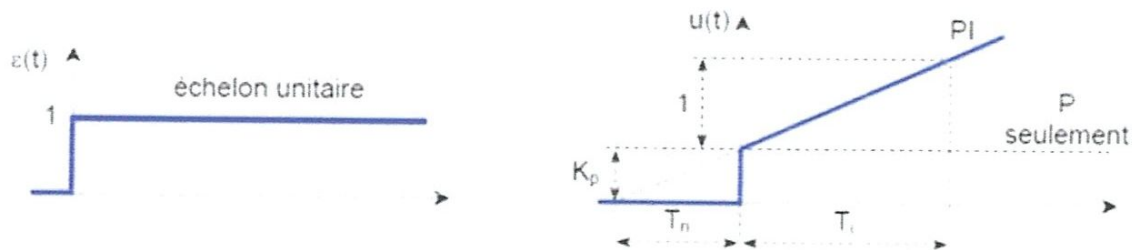


Figure I.14 : Entrée et sortie du correcteur PI

I.6.3.2 Effet :

La réponse indicielle montre qu'un correcteur PI assure une transmission instantanée du signal d'erreur, suivi d'une intégration de ce signal.

- Ce correcteur sera utilisé chaque fois qu'une erreur permanente doit être annulée ou minimisée, c'est à dire une amélioration de la précision du système. En effet, il introduit une augmentation du gain global du système aux basses fréquences.
- Par ailleurs, le correcteur PI a un effet déstabilisant en raison du pôle à l'origine (déphasage supplémentaire entre 0 et -90°). Mais, Le zéro supplémentaire introduit tend à minimiser cette instabilité.
- Il est recommandé de placer le zéro $\frac{K_i}{K_p}$ du correcteur aux basses fréquences

De sorte que le déphasage supplémentaire introduit par le correcteur n'affecte pas beaucoup le déphasage global du système corrigé. Cependant, s'il est très proche de l'origine, son effet sera compensé par le pôle à l'origine.

➤ K_p sera choisi de manière à modifier, éventuellement, la fréquence de coupure du système corrigé et donc sa marge de phase.

- Très souvent, le zéro $\frac{K_i}{K_p}$ est choisi de manière à compenser la constante de

Temps dominante du système initial de sorte que la boucle fermée gagne en rapidité.

K_p et K_i sont tous deux réglables. K_i ajuste l'action intégrale, tandis que K_p affecte à la fois les actions intégrale et proportionnelle.

I.6.4 Correcteur à Action Dérivée (D) :**I.6.4.1 Principe :**

La relation entre la sortie $u(t)$ et le signal d'erreur $\varepsilon(t)$ est :

$$u(t) = K_d \cdot \frac{d\varepsilon(t)}{dt}$$

Sa fonction de transfert est donc :

$$C(p) = \frac{U(p)}{\varepsilon(p)} = K_d \cdot p = T_d \cdot p$$

Où :

K_d : gain dérivé,

T_d : constante de temps de dérivation.

Son schéma fonctionnel est celui de la figure I.15.

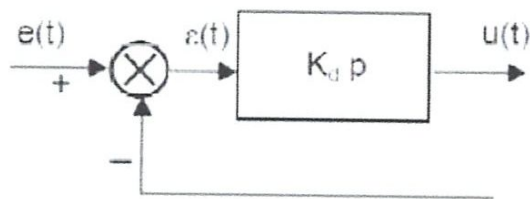


Figure I.15 : Correction à action dérivée pure (D)

La figure I.16 donne les réponses à un échelon et à une rampe du correcteur D.

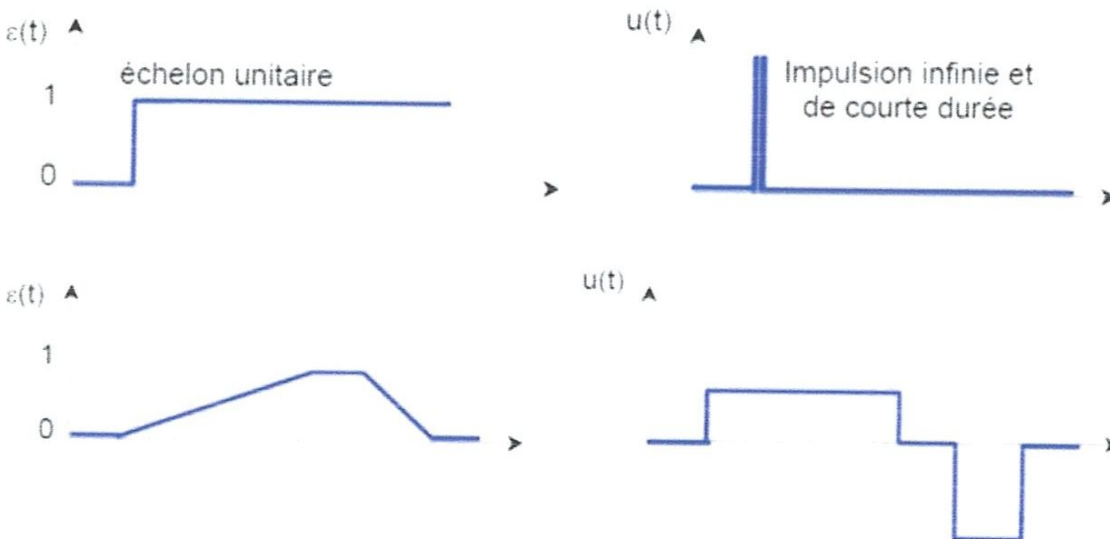


Figure I.16 : Entrée et sortie du correcteur à action dérivée (D)

I.6.4.2 Effet :

La réponse indicielle montre qu'un correcteur à action exclusivement dérivée ne permet pas la transmission d'un signal L'action dérivée ne peut donc être utilisée seule. On fait appel à elle lorsque le signal de commande u doit être particulièrement efficace. En effet, ce correcteur permet de faire intervenir la dérivée du signal d'erreur; il sera d'autant plus actif que la variation de $\varepsilon(t)$ est rapide.

L'action dérivée pure :

- améliore la stabilité du système par l'introduction d'un déphasage supplémentaire de + 90° (augmentation de la marge de phase),
- mais fait diminuer la précision du système,
- et amplifie les bruits de hautes fréquences.

Le correcteur à action exclusivement dérivée n'est pratiquement jamais utilisé. Il est en général associé au correcteur Proportionnel.

I.6.5 Correcteur à Actions Proportionnelle et Dérivée (PD) :**I.6.5.1 Principe :**

La relation entre la sortie $u(t)$ et le signal d'erreur $\varepsilon(t)$ est :

$$u(t) = K_p \cdot \varepsilon(t) + K_d \cdot \frac{d\varepsilon(t)}{dt}$$

Sa fonction de transfert est donc :

$$C(p) = \frac{U(p)}{\varepsilon(p)} = K_p + K_d \cdot p = K_p \left(1 + \frac{K_d}{K_p} \cdot p \right) = K_p (1 + T_d p)$$

Où :

K_p : gain proportionnel.

K_d : gain dérivé,

T_d : constante de temps de dérivation.

Son schéma fonctionnel est celui de la figure I.17.

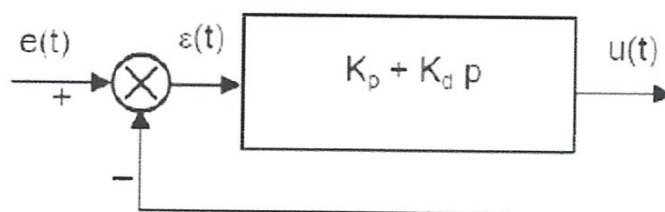


Figure I.17 : Correction PD

La figure I.18 donne les réponses à un échelon et à une rampe.

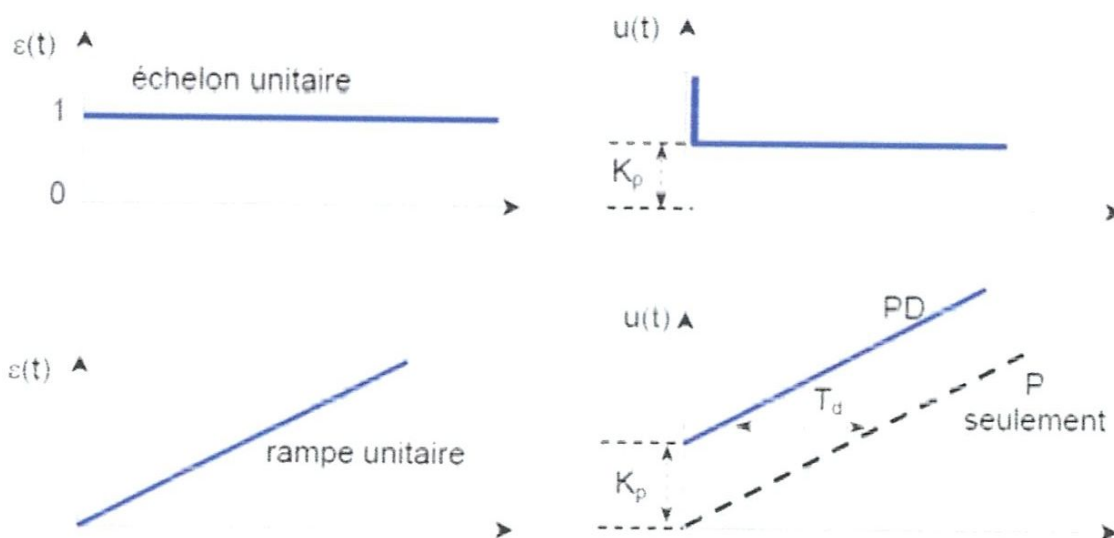


Figure I.18 : Entrée et sortie du correcteur PD

I.6.5.2 Effet :

- Amélioration de l'amortissement et réduction du dépassement.
- Réduction du temps de montée et du temps d'établissement (temps de réponse).
- Augmentation de la bande passante.
- Amélioration de la marge de phase et de la marge de gain.
- Possibilité d'accentuation des bruits aux hautes fréquences.

I.6.6 Correcteur à Actions Proportionnelle, Intégrale et Dérivée (PID) :

I.6.6.1 Principe :

Le correcteur PID combine les actions des 3 correcteurs P, I et D.

La relation entre la sortie $u(t)$ et le signal d'erreur $\varepsilon(t)$ est :

$$u(t) = K_p \cdot \varepsilon(t) + K_i \cdot \int_0^t \varepsilon(t) dt + K_d \cdot \frac{d\varepsilon(t)}{dt}$$

Sa fonction de transfert est donc :

$$C(p) = \frac{U(p)}{\varepsilon(p)} = K_p + \frac{K_i}{p} + K_d \cdot p = K_p \left(1 + \frac{K_d}{K_p} \cdot p + \frac{K_i}{K_p} \frac{1}{p} \right) = K_p \left(1 + T_d \cdot p + \frac{1}{T_i p} \right)$$

Où :

K_p : gain proportionnel.

K_i : gain intégral.

K_d : gain dérivé,

$$T = \frac{K_d}{K_p} : \text{constante de temps de dérivation.}$$

$$T = \frac{K_p}{K_i} : \text{constante de temps d'intégration.}$$

Son schéma fonctionnel est celui de la fig. I.19.

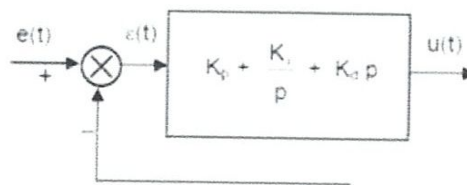


Figure I.19 : Correction à actions proportionnelle, intégrale et dérivée (PID)

La figure I.20 donne les réponses à un échelon et à une rampe.

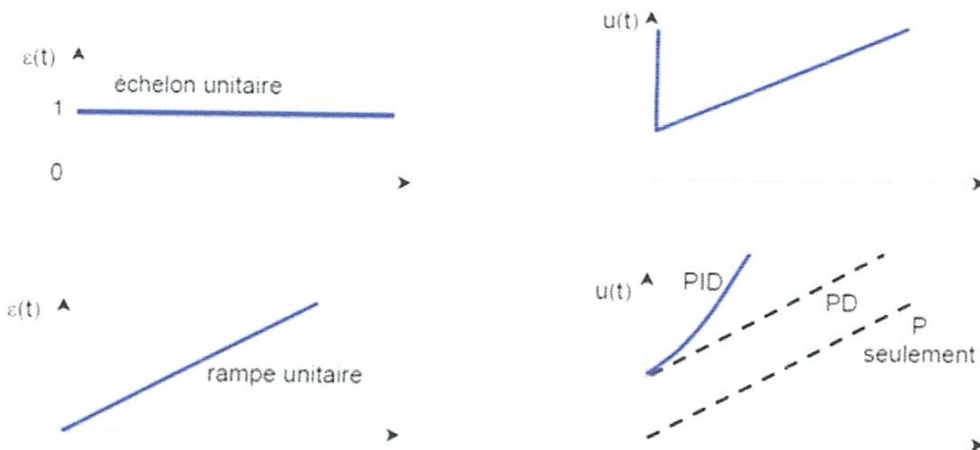


Figure I.20 : Entrée et sortie du correcteur à actions proportionnelle, intégrale et dérivée (PID)

I.6.6.2 Effet :

Les effets de chaque correcteur (K_p , K_i et K_d) sur la réponse en boucle fermée du système sont regroupés sur le tableau I.1 :

	Temps de montée	Dépassement	Temps d'établissement	Erreur statique
Si K_p croit	Diminue	Augmente	(Peu de changements)	Diminue
Si K_i croit	Diminue	Augmente	Augmente	Eliminée
Si K_d croit	(Peu de changements)	Diminue	Diminue	(Peu de changements)

Tableau I.1 : Effets des correcteurs P, I, et D sur les régimes statique et dynamique du système en boucle fermée.

Il est à noter que ces corrélations ne sont pas exactement précises, car K_p , K_i et K_d sont dépendants les uns des autres. En fait, le changement de l'une de ces variables peut modifier l'effet de l'autre. Le tableau précédent n'est à utiliser que comme référence lorsqu'il s'agit de déterminer les valeurs de K_p , K_i et K_d [3].

Il existe des méthodes analytiques permettant de calculer les composantes du correcteur PID, mais elles sont assez complexes et sont peu utilisées. Des méthodes empiriques existent et

permettent de faciliter amplement la détermination du correcteurs PID (méthode de Ziegler-Nichols, méthode de Chien-Hrones-Reswick,...).

Des méthodes pratiques de réglages permettent d'obtenir de bons résultats. Elles sont basées sur la connaissance des effets que procure chaque correcteur sur la réponse du système bouclé (tableau I.1). Par ailleurs, ces méthodes font beaucoup intervenir l'expérience de l'opérateur dans ce domaine. Il n'y a pas de réglage unique permettant d'atteindre le cahier des charges, mais il est nécessaire de suivre quelques règles d'ajustement de ces correcteurs :

- 1) *Obtenir la réponse en boucle ouverte et déterminer ce qui est nécessaire d'améliorer.*
- 2) *Rajouter un correcteur P pour améliorer la rapidité du système : modifier K_p pour obtenir le temps de montée voulu.*
- 3) *Rajouter un correcteur I pour éliminer l'erreur statique : modifier K_i pour améliorer les performances en régime statique.*
- 4) *Rajouter un correcteur D pour réduire les dépassements et améliorer le temps d'établissement : modifier K_d pour améliorer les caractéristiques en régime transitoire.*
- 5) *Ajuster K_p , K_i et K_d jusqu'à obtenir les performances voulues.*

Finalement, il faut se rappeler qu'il n'est pas obligatoire d'insérer les 3 correcteurs dans un même système si cela n'est pas nécessaire. Si un correcteur PI donne des performances satisfaisantes pour la sortie, il n'est alors pas nécessaire de rajouter un correcteur D au système. Construire le correcteur aussi simplement que possible.

I.7 Conclusion :

Dans ce chapitre nous avons présenté le rôle des correcteurs dans un système de régulation pour la réalisation d'un meilleur compromis entre la précision, Stabilité et rapidité du système. Ainsi que l'étude fréquentielle de chaque correcteur dans le but de déterminer l'intervalle de son fonctionnement, Le régulateur standard le plus utilisé dans l'industrie est le régulateur PID (Proportionnel intégral dérivé) car il permet de régler à l'aide de ses trois paramètres les performances (Amortissement, Temps de réponse) d'une régulation d'un processus.

Chapitre II :
La logique floue

II .Introduction :

Les méthodes conventionnelles de réglage sont basées sur une modélisation adéquate du système à régler et un traitement analytique à l'aide de la fonction de transfert ou d'équations d'état. Malheureusement, celles-ci ne sont pas toujours disponibles.

Les techniques de commande ont prouvé leur efficacité dans de nombreux problèmes de régulation industrielle. Les méthodes de commande avancées (Régulateur adaptatif, Commande prédictive, Commande robuste...) permettent de répondre aux exigences d'un certain nombre de systèmes fortement non linéaires. C'est dans ce même créneau que les méthodes de modélisation et de commande floues se positionnent [4].

La majorité des systèmes industriels complexes sont difficiles à contrôler automatiquement. Cette difficulté provient de :

- Leur non-linéarité,
- La variation de leurs paramètres,
- La qualité des variables mesurables.

Ces difficultés ont conduit à l'avènement et au développement de nouvelles techniques telles que la commande floue particulièrement intéressante lorsqu'on ne dispose pas de modèle mathématique précis du processus à commander ou lorsque ce dernier présente de forts non linéarités ou imprécisions [5].

Dans plusieurs applications, les résultats obtenus avec un contrôleur flou sont meilleurs que ceux obtenus avec un algorithme de contrôle conventionnel. En particulier, la méthodologie du contrôle flou apparaît utile quand les processus sont très complexes à analyser par les techniques conventionnelles. Plusieurs travaux dans le domaine de commande ont montré qu'un régulateur par logique floue est plus robuste qu'un régulateur conventionnel.

Nous présentons dans ce chapitre les concepts de la logique floue et nous décrivons les aspects méthodologiques nécessaires à la compréhension de cette technique en donnant des rappels sur les bases générales de la logique floue.

II.1 Historique de la logique floue :

Le terme d'ensemble flou apparaît pour la première fois en 1965 lorsque le professeur **Lotfi A. Zadeh**, de l'université de Berkeley aux USA [4], publie un article intitulé « Ensembles flous » (**Fuzzy sets**). Il a réalisé depuis de nombreuses avancées théoriques majeures dans le domaine et a été rapidement accompagné par de nombreux chercheurs développant des travaux théoriques

La logique floue suscite actuellement un fervent intérêt de la part des chercheurs, des ingénieurs et des industriels, mais plus généralement de la part de tous ceux qui éprouvent le besoin de formaliser des méthodes empiriques, de généraliser les modes de raisonnement naturels, d'automatiser la prise de décision dans leur domaine et de construire des systèmes artificiels capables d'effectuer les tâches habituellement prises en charge par les humains.

Les démarches fondamentales de l'activité industrielle et économique, telles que la conception des produits, la gestion des systèmes ou la prise de décision, posent des problèmes de complexité croissante. Face à cette difficulté, les approches numériques, pourtant bien développées se sont avérées d'une efficacité limitée. **Zadeh** a fait le lien entre les ensembles flous et la modélisation non probabiliste de l'incertitude grâce à la théorie des possibilités, dans laquelle les fonctions d'appartenance des ensembles flous servent à représenter les idées de valeur incomplètement spécifiée et mal connue. Les ensembles flous et la théorie des possibilités sont à la base d'une méthodologie de représentation et d'utilisation de connaissances imprécises, vagues et incertaines que **Zadeh** appelle «théorie du raisonnement approché » et qui est plus connue sous le vocable de « logique floue» [5].

Les travaux de Lotfi **Zadeh** dans les années soixante sur la théorie des ensembles flous et la logique floue, à l'université de Berkeley ont jeté les bases de la commande floue actuelle. Ces travaux ont mis un certain temps avant d'être acceptés par la communauté de l'automatique et l'industrie. En effet la logique floue, de part ses aptitudes pluridisciplinaires (utilisée dans des domaines aussi variés que la gestion de base de données, les systèmes experts, le traitement du signal, la classification de données, la commande de systèmes automatiques), s'est attirée les critiques des scientifiques qui se sentaient attaqués sur leurs positions [4], il fallait attendre les années quatre-vingts dix pour que ces nouvelles théories soit acceptées et utilisées d'une manière générale.

En revanche, dans le domaine de la commande, la première application de la logique floue a été réalisée par Mamdani en 1974 [5]. Depuis 1987, date la première application utilisant la logique floue, le métro de Sendai, l'utilisation des techniques de commande floue n'a cessé de s'accroître [4].

De nos jours, la logique floue (fuzzy logic) est un axe de recherche important sur lequel se focalisent de nombreux scientifiques. Des retombées technologiques sont d'ores et déjà disponibles, tant dans le domaine grand public (appareils photos, machines à laver, fours à micro-onde), que dans le domaine industriel (classification, aide à la décision, réglage et commande de processus, complexes liés à l'énergie, aux transports, à la transformation de la matière, à la robotique, aux machines-outils).

La logique floue, dans la plupart des applications actuelles, permet de prendre en compte toutes sortes de connaissances qualitatives de concepteurs et d'opérateurs dans l'automatisation des Systèmes.

Elle suscite en France un intérêt médiatique depuis le début des années 90. Les nombreuses Applications dans l'électroménager et l'électronique grand public réalisées notamment au Japon en ont été l'élément déclenchant.

Machines à laver sans réglage, caméscopes anti-bougé et de nombreuses autres innovations ont fait connaître le terme « logique floue » à un large public. Dans l'automobile les transmissions automatiques, les contrôles d'injection et d'anti cliquetis, l'air conditionné sont réalisés sur des véhicules de série grâce à la logique floue.

Dans le domaine des processus de production, continue et par lots, et dans les automatismes (qui nous intéressent essentiellement ici) les applications se sont également multipliées. La logique floue s'y développe car il s'agit d'une approche essentiellement pragmatique, efficace et générique. On dit parfois qu'elle permet de systématiser ce qui est du domaine de L'empirisme, et donc difficile à maîtriser. La théorie des ensembles flous fournit une méthode Pertinente et facilement réalisable dans des applications temps réel ; elle permet de transcrire et rendre dynamiques les connaissances des concepteurs ou des opérateurs.

Cet aspect adaptable et universel de la logique floue permet de s'attaquer à l'automatisation de Procédures telles que la mise en route, le réglage de paramètres, pour lesquelles peu d'approches existaient auparavant.

II.2 Principe de la Logique Floue :

Dans le cadre de la logique classique, une proposition est soit vraie, soit fausse (1 ou 0). 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». La figure 1.1a montre le résultat de cette partition. 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. La logique floue permet de définir des sous-ensembles, comme «froid» ou «chaud», en introduisant la possibilité pour une valeur d'appartenir plus ou moins à chacun de ces sous-ensembles [6].

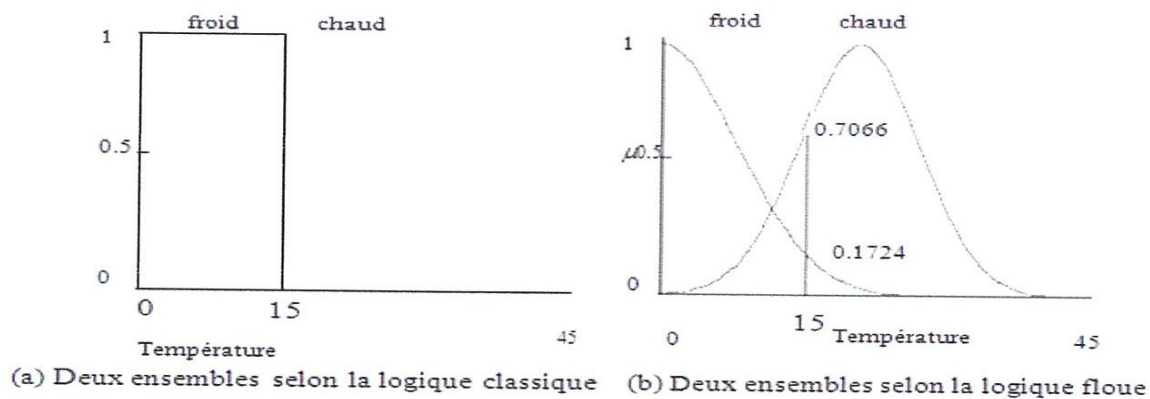


Figure II.1 : classification des températures d'une pièce en deux ensembles.

La logique floue permet de définir des sous-ensembles, comme «froid» ou «chaud», en introduisant la possibilité pour une valeur d'appartenir plus ou moins à chacun de ces sous-ensembles.

Exemple 1 :

Pour bien mettre en évidence le principe fondamental de la logique floue, on présente un simple exemple : Comment classer des personnes en trois ensembles : "Jeunes", "entre 2 âges" et "âgé".

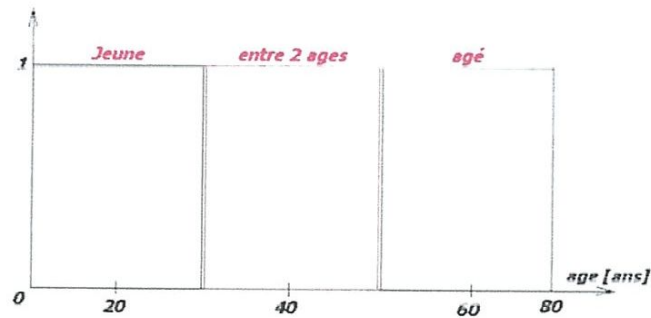


Figure II.2 : Classification selon la logique classique

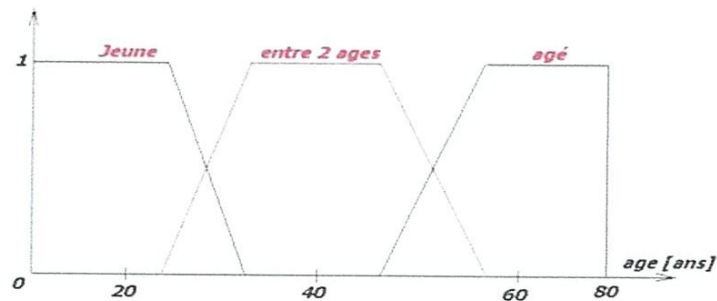


Figure II.3 : Classification selon la logique floue

Selon ces deux figures, la classification des personnes en trois ensembles est bien clair mais très différentes, en effet :

Selon la logique classique (logique de Boole), toutes personnes âgées de moins de 30 ans sont considérées comme "jeune" et celles âgées de 50 ans appartiennent l'ensemble âgé or, en vérité le passage du jeune au vieux ce fait progressivement et individuellement ce qui nous amène à dire qu'une telle classification n'est pas logique.

Cependant, la logique floue, dont les variables peuvent prendre n'importe quelle valeur. On parlera de fonction d'appartenance Y , permet de tenir compte du fait qu'une personne de 25 ans est considéré comme jeune avec un certain pourcentage ($Y=0,75\%$), et comme étant entre les 2 âges avec un autre % ($Y =0,25\%$).

II.3 Sous-ensembles flous :

Dans cette section, nous décrivons rapidement les fondements mathématiques de la théorie des sous-ensembles flous [7]. Dans la théorie ensembliste classique, 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.

II.3.1 Définitions :

Etant donné un ensemble de référence X qui peut être fini ou infini, dénoté par ses éléments $\{x\}$, on peut indiquer les éléments $\{x\}$ qui appartiennent à une certaine classe de X (on leur donne une valeur 1) et ceux qui n'y appartiennent pas (on leur donne une valeur 0). Cette classe est alors un sous-ensemble classique de X caractérisé par une fonction caractéristique \mathcal{A}_i prenant simplement deux valeurs 0 ou 1 :

$$\mathcal{A}_i : X \rightarrow \{0, 1\} \quad (2.1)$$

Si l'appartenance de certains éléments de X à une classe n'est pas absolue (l'élément appartient *un peu* au sous-ensemble), on peut remplacer la fonction caractéristique par une *fonction d'appartenance* qui prend ses valeurs dans l'intervalle $[0, 1]$. Cette classe est appelée sous-ensemble flou de X . L'ensemble X sera également appelé *univers du discours*.

Définition 1.1 (Sous-ensemble flou) :

Un sous-ensemble flou A dans un univers du discours X est caractérisé par sa fonction d'appartenance $\mu_A(x)$ qui associe à chaque élément x de X une valeur dans l'intervalle des nombres réels $[0, 1]$.

$$\mu_A : X \rightarrow [0, 1]. \quad (2.2)$$

Ainsi un sous-ensemble flou A dans X peut être représenté par un ensemble de couples ordonnés

$$A = \{(x, \mu_A(x)) | x \in X\}. \quad (2.3)$$

Le sous-ensemble classique n'est en fait qu'un cas particulier de sous-ensemble flou dont la fonction d'appartenance ne prend que les valeurs 0 ou 1. Un sous-ensemble flou A de X est aussi souvent représenté par la notation suivante qui indique pour tout élément x de X son degré $\mu_A(x)$ d'appartenance à A :

$$A = \int_X \mu_A(x) / x \quad \text{si } X \text{ est continu} \quad (2.4)$$

Et

$$A = \sum_{x_i \in X} \mu_A(x_i) / x_i \quad \text{si } X \text{ est discret} \quad (2.5)$$

Comme les valeurs $\mu_A(x_i)$ représentent les degrés d'appartenance avec lesquels les x_i appartiennent à A , si $\mu_A(x_i)$ prend la valeur 1 pour tous les éléments de X , cela signifie que A est identique à X . Au contraire, A est vide si $\mu_A(x_i)$ prend la valeur 0 sur tout X .

En commande floue, les fonctions d'appartenance utilisées peuvent théoriquement être quelconques. Pourtant on choisit souvent des fonctions triangulaires ou trapézoïdales afin de simplifier les calculs.

Définition 1.2 (Support) :

Le support d'un sous-ensemble flou A dans un univers du discours X est le sous-ensemble (au sens classique du terme) des éléments de X pour lesquels la fonction d'appartenance prend une valeur strictement positive. C'est l'ensemble des éléments de X qui appartiennent au moins un peu à A :

$$S(A) = \{x | \mu_A(x) > 0\}. \quad (2.6)$$

Définition 1.3 (Point de croisement) :

Le point de croisement d'un sous-ensemble flou A dans un univers du discours X est le sous-ensemble des éléments de X pour lesquels la fonction d'appartenance prend une valeur égale à 0.5. C'est l'ensemble des éléments de X qui appartiennent autant à A qu'à son complémentaire :

$$C(A) = \{x | \mu_A(x) = 0.5\}. \quad (2.7)$$

Définition 1.4 (Noyau) :

Le noyau d'un sous-ensemble flou A dans un univers du discours X est le sous-ensemble des éléments de X pour lesquels la fonction d'appartenance vaut 1. C'est l'ensemble des points qui appartiennent intégralement à A :

$$N(A) = \{x | \mu_A(x) = 1\}. \quad (2.8)$$

Définition 1.5 (Hauteur) :

La hauteur d'un sous-ensemble flou A dans un univers du discours X est la valeur maximale prise par la fonction d'appartenance μ_A sur l'ensemble X . C'est le plus fort degré avec lequel un élément de X appartient à A :

$$H(A) = \sup_{x \in X} \mu_A(x). \quad (2.9)$$

II.4 Différentes formes des fonctions d'appartenance :

Les fonctions d'appartenance, leurs positions ainsi que leurs chevauchements peuvent être choisis librement. Elles (les fonctions d'appartenance) peuvent être symétriques, régulièrement distribuées ou avoir une distribution non uniforme, cependant, il convient de respecter certaines règles [8]:

- Leur forme doit être convexe,
- Les sous-ensembles d'entrées (souvent des trapèzes et des triangles) doivent se chevaucher partiellement afin qu'il n'y ait pas de zones indéterminées, tandis que pour ceux de sorties (souvent des trapèzes et des triangles), cette précaution n'est pas nécessaire,

• Éviter d'imbriquer plus de deux sous-ensembles, sinon il y aura des problèmes d'instabilité. On peut choisir des formes quelconques de fonctions d'appartenance: une forme de cloche, trapézoïdale, ou triangulaire. Mais le plus souvent on choisira une forme triangulaire ou trapézoïdale car formée de segments de droite, le calcul est plus rapide, entre autres, le calcul de la surface de la fonction. Pour la variable "Environ zéro", on doit éviter de prendre une forme trapézoïdale susceptible d'entraîner des erreurs de réglage [6] [9] [8].

Mathématiquement, la fonction d'appartenance μ comprise entre 0 et 1 est associée aux sous-ensembles E de $X(T)$ et à la variable x , dont l'ensemble des valeurs possible est $X(T)$.

Elle s'appelle aussi, degré d'appartenance ou coefficient d'appartenance ou degré de possibilité, qui est la possibilité que la variable u ait la qualité associée au sous-ensemble E .

Elle est utilisée pour le traitement mathématique des variables linguistiques dans le but de traiter des déductions floues par ordinateur [10]. Ils existent plusieurs formes de la fonction d'appartenance dont ces fonctions sont les suivantes:

a) fonction d'appartenance triangulaire

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

b) fonction d'appartenance trapézoïdale

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

c) fonction d'appartenance gaussienne

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

La figure 5 représente les formes de ces trois types de fonction d'appartenance

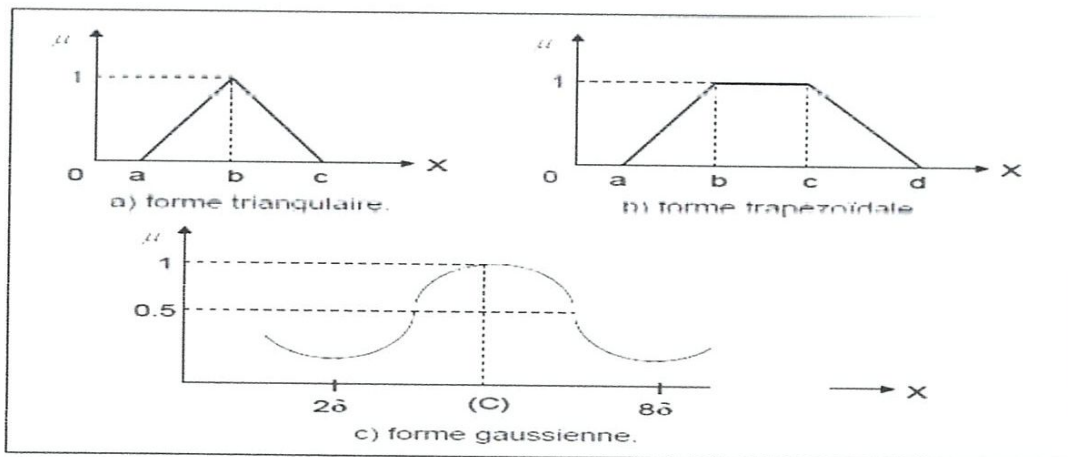


Figure II.4 : Différentes formes de la fonction d'appartenance

II.4.1 Opérations sur les sous-ensembles flous :

Supposons que A et B sont deux sous-ensembles flous définis dans un univers du discours X par les fonctions d'appartenance μ_A et μ_B . On peut définir des opérations ensemblistes telles que l'égalité, l'inclusion, l'intersection, l'union et le complément grâce à des opérations sur les fonctions d'appartenance.

Définition 1.6 (Égalité) : A et B sont dits égaux, propriété que l'on note $A = B$, si leurs fonctions d'appartenance prennent la même valeur en tout point de X :

$$\forall x \in X \quad \mu_A(x) = \mu_B(x). \tag{2.10}$$

Définition 1.7 (Inclusion) : A est dit inclus dans B, propriété que l'on note $A \subseteq B$, si tout élément x de X qui appartient à A appartient aussi à B avec un degré au moins aussi grand :

$$\forall x \in X \quad \mu_A(x) \leq \mu_B(x). \tag{2.11}$$

Les définitions d'intersection, d'union et de complément de sous-ensembles flous définies ci-dessous font intervenir les opérateurs de minimum, maximum et de complémentation à 1. Cela correspond à une extension triviale des opérateurs ensemblistes standards.

Définition 1.8 (Intersection) :

L'intersection de A et B, que l'on note $A \cap B$, est le sous-ensemble flou constitué des éléments de X affectés du plus petit des deux degrés d'appartenance μ_A et μ_B :

$$\forall x \in X \quad \mu_{A \cap B} = \min(\mu_A(x), \mu_B(x)) = \mu_A(x) \wedge \mu_B(x). \tag{2.12}$$

Dans cette définition, min et \wedge désignent l'opérateur de calcul du minimum des deux valeurs.

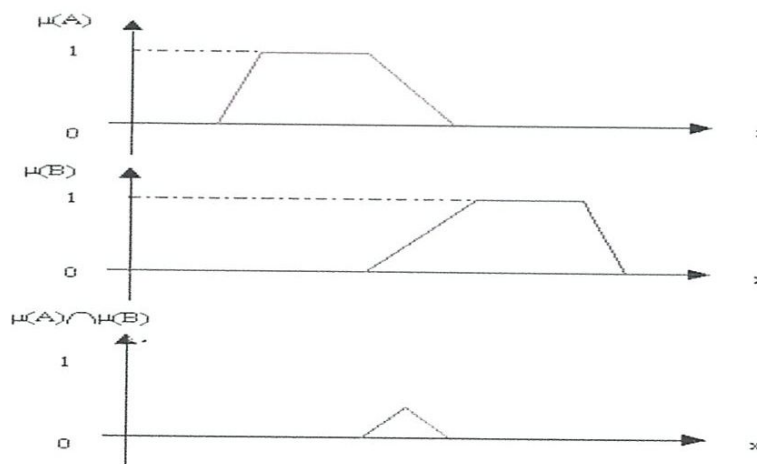


Figure II.5 : Intersection des fonctions d'appartenance

Définition 1.9 (Union) : L'union de A et B , que l'on note $A \cup B$, est le sous-ensemble flou constitué des éléments de X affectés du plus grand des deux degrés d'appartenance μ_A et μ_B :

$$\forall x \in X \quad \mu_{A \cup B} = \max(\mu_A(x), \mu_B(x)) = \mu_A(x) \vee \mu_B(x). \quad (2.13)$$

\vee désignent l'opérateur de calcul du maximum de deux valeurs.

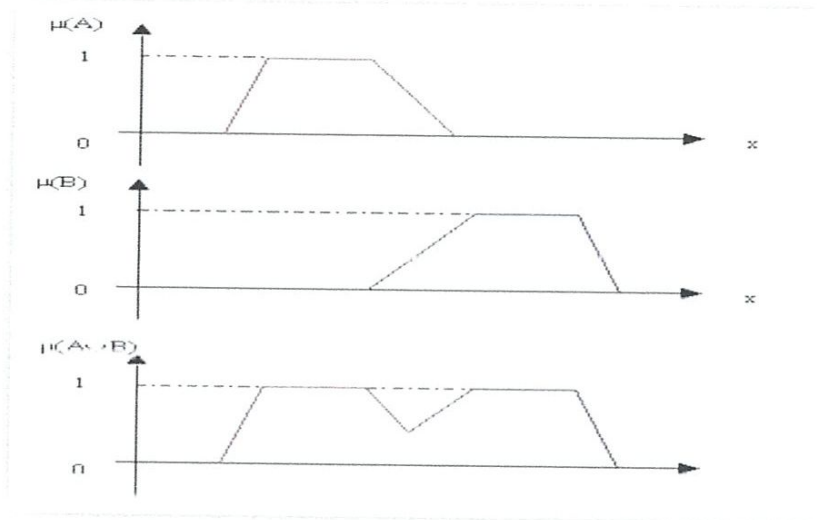


Figure II.6 : Union des fonctions d'appartenances.

Définition 1.10 (Complément) :

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

$$\forall x \in X \quad \mu_{A^c} = 1 - \mu_A(x). \quad (2.14)$$

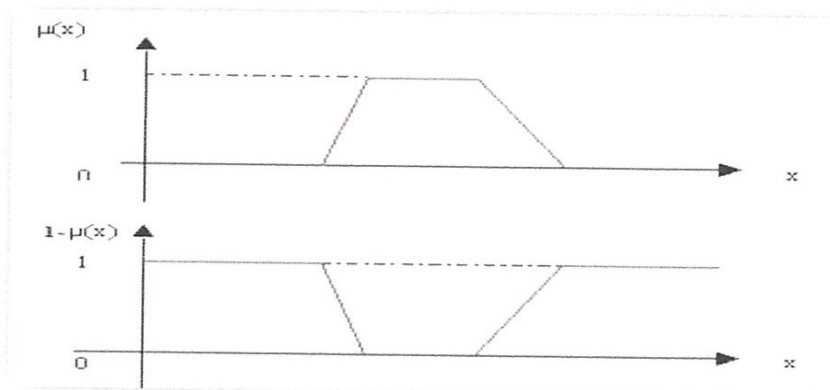


Figure II.7 : Fonction d'appartenances.

II.4.2 Normes et conormes triangulaires :

Comme nous l'avons vu dans le paragraphe précédent, la définition d'une opération entre ensembles flous est basée sur une combinaison des fonctions d'appartenance. Les définitions les plus simples utilisent les opérations de minimum, maximum et de complément à 1. Les normes et conormes triangulaires constituent une généralisation des opérations de combinaison de type minimum ou maximum.

Définition 1.11 (Norme triangulaire, t-norme) :

Une norme triangulaire T est une fonction définie sur l'ensemble $[0, 1] \times [0, 1]$ et prenant ses valeurs dans l'intervalle $[0, 1]$, qui satisfait les conditions suivantes :

$$\forall x, y \in [0, 1] : \quad T(x, y) = T(y, x) \text{ (commutativité),}$$

$$\forall x, y, z \in [0, 1] : \quad T(x, T(y, z)) = T(T(y, z), z) \text{ (associativité),}$$

$$\forall x, y, z, t \in [0, 1] : \quad T(x, y) \leq T(z, t) \text{ si } x \leq z \text{ et } y \leq t \text{ (monotonie),}$$

$$\forall x \in [0, 1] : \quad T(0, 0) = 0, \quad T(x, 1) = T(1, x) = x \text{ (élément neutre 1).}$$

II.4.3 Relations floues :

Dans la section précédente, nous avons présenté le produit cartésien de deux sous-ensembles flous. Cette méthode permet de construire un sous-ensemble flou sur un univers du discours qui est lui-même un produit cartésien. Dans l'exemple présenté précédemment sur un univers du discours température et hygrométrie, le produit cartésien n'est assurément pas une bonne méthode pour définir le sous-ensemble flou chaud et humide : en effet, les grandeurs initiales ne sont pas indépendantes.

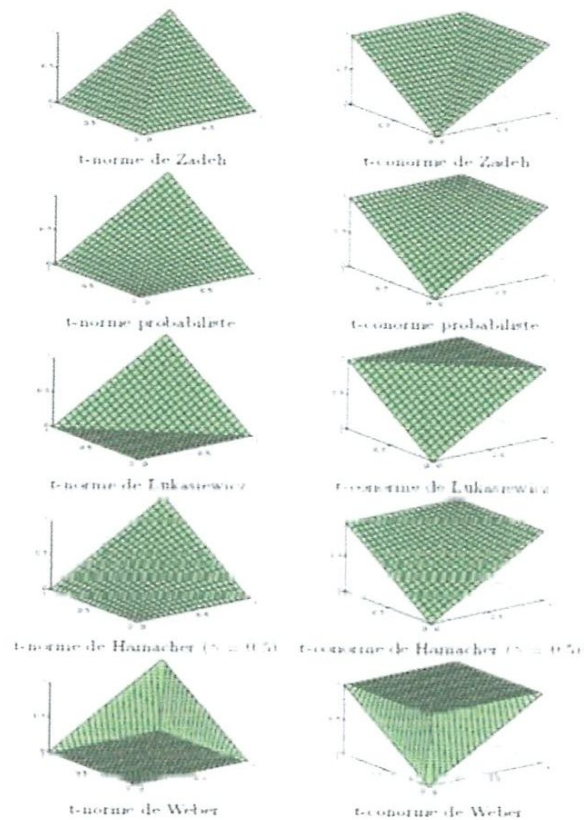


Figure II.8 : t-normes et t-conormes les plus utilisées [7]

II.5 Raisonnement en logique floue :

Un des apports principaux de la logique standard a été la formalisation des méthodes de déduction, qui sont en quelque sorte un outil de raisonnement. Les méthodes de déduction utilisées en logique standard permettent de définir une nouvelle certitude à partir d'autres connaissances certaines. Dans le cadre de la logique floue, il est possible de généraliser les méthodes de raisonnement lorsqu'on dispose de connaissances incertaines ou imprécises.

II.5.1 Variables linguistiques :

Pour qu'il soit possible de raisonner simplement sur un problème, il faut tout d'abord spécifier clairement les connaissances disponibles. Les variables linguistiques permettent de décrire dans un cadre très général la connaissance acquise sur une variable, même lorsqu'elle est vague ou imprécise [11].

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

X : le nom de variable

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

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

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

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

Pour les vitesses inférieures ou égales à 80 intervient la variable linguistique (lente), et pour les vitesses supérieures ou égales à 150 intervient la variable linguistique (rapide). Ces deux termes peuvent être caractérisés comme des ensembles flous dont les fonctions d'appartenance sont montrées sur La figure (II.9)

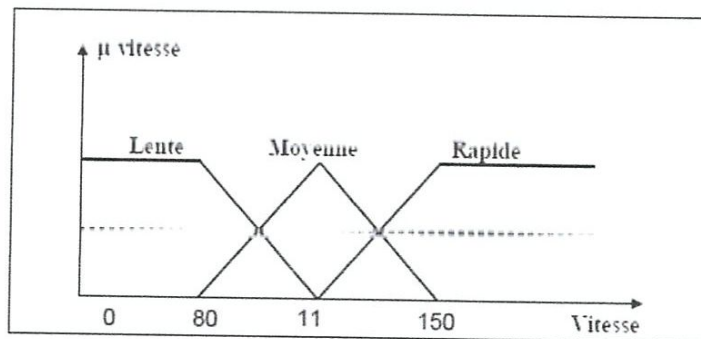


Figure II.9 : Représentation des variables linguistiques

Exemple 2 :

Afin de décrire la température d'une pièce par une variable linguistique, on peut utiliser l'ensemble des termes suivants : $T(V) = \{\text{froid, tiède, chaud}\}$. D'autres termes linguistiques peuvent être formés en utilisant le modificateur linguistique «très», comme «très froid» ou «très très chaud». En considérant que l'univers du discours est l'intervalle $[0, 45]$, on peut utiliser les règles sémantiques suivantes pour définir les termes linguistiques :

«froid» est «une température environ inférieure à 10 degrés», «tiède» est «une température d'environ 17 degrés» et «chaud» est «une température environ supérieure à 24 degrés». Ces termes peuvent être caractérisés par les fonctions d'appartenance représentées sur la figure (II.10) N'importe quelle fonction d'appartenance, par exemple la fonction «froid», définit un nombre flou.

II.5.2 Propositions floues :

Une proposition floue est définie à partir d'un ensemble de variables linguistiques afin de représenter une connaissance. Par exemple, «la température de la pièce est froide».

Définition 1.12 (Propositions floues élémentaires) :

Une forme élémentaire de proposition floue est définie à partir d'une seule variable linguistique $(V, T(V), X, G, M)$ et exprimée simplement par la phrase :

$$p : V \text{ est } A$$

Où v est une variable qui prend sa valeur dans l'univers du discours X , et A est l'un des termes linguistiques de $T(V)$. Une valeur particulière $V = v$ appartient à A avec le degré d'appartenance $\mu_A(v)$. Cela permet de définir la valeur de vérité $V(p)$ de la proposition lorsque v vaut v :

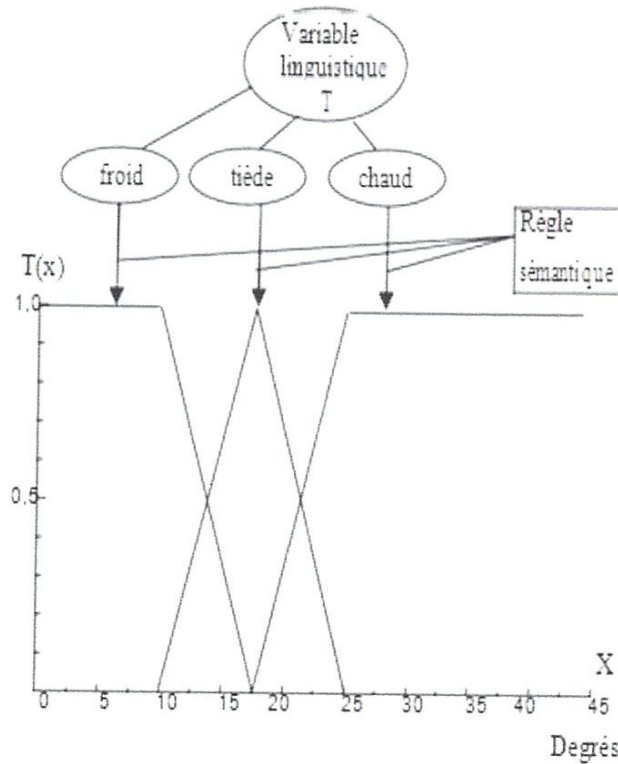


Figure II.10 : Variable linguistique

Exemple 3 :

Supposons que l'on veuille définir la température V d'une pièce en utilisant la variable linguistique définie dans l'exemple 2. La proposition floue correspondante est exprimée par la phrase :

$$p : V \text{ est tiède.}$$

A partir de la fonction d'appartenance définissant «tiède», nous trouvons que le degré d'appartenance d'une température de 15 degrés au sous-ensemble flou «tiède» est de 0.7, comme on le voit sur la figure 1.7a. La valeur de vérité $V(p)$ de la proposition p lorsque la température vaut 15 degrés est donc de 0.7.

Nous voyons clairement que le rôle de la fonction $V(p)$ est de définir un lien entre les sous-ensembles flous et les propositions floues. Comme la valeur de vérité $V(p)$ est exprimée à partir de la fonction d'appartenance $\mu_A(x)$ dans l'équation 1.30 et qu'elle est introduite pour décrire clairement une connaissance, nous remplaçons directement la valeur de vérité $V(p)$ par la fonction d'appartenance $\mu_A(x)$.

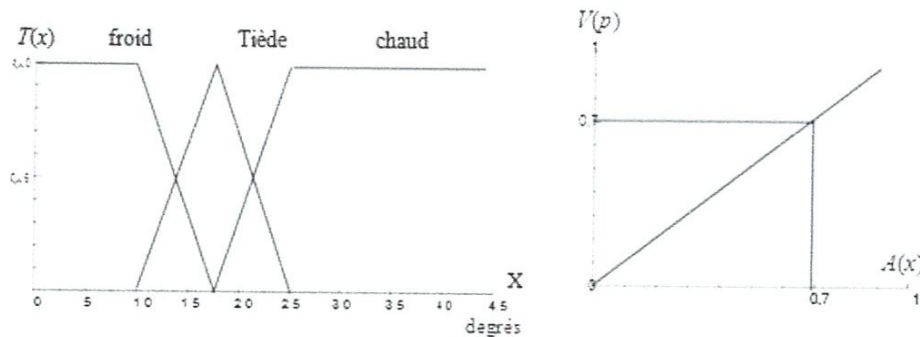


Figure II.11 : Valeur de vérité de la proposition floue p : la température est tiède

Définition 1.13 (Propositions floues générales) :

Une proposition floue générale est obtenue par la composition de propositions élémentaires « x est A », « y est B »,... pour des variables x, y, \dots supposées non indépendantes.

Habituellement les propositions floues générales sont classées en quatre types :

- La conjonction de propositions floues élémentaires :

$$p : (\mathcal{X}_1 \text{ est } A_1) \text{ et } \dots \text{ et } (\mathcal{X}_n \text{ est } A_n).$$

Dans ce cas, la conjonction est associée au produit cartésien $A_1 \times A_2 \times \dots \times A_n$

caractérisant la variable conjointe $(\mathcal{X}_1, \dots, \mathcal{X}_n)$ sur les univers de discours $X_1 \times X_2 \times \dots \times X_n$. Sa valeur de vérité est alors définie par :

$$V(p) = \min\{\mu_{A_1}(\mathcal{X}_1), \dots, \mu_{A_n}(\mathcal{X}_n)\}$$

- La disjonction de propositions floues élémentaires :

$$p : (\mathcal{X}_1 \text{ est } A_1) \text{ ou } \dots \text{ ou } (\mathcal{X}_n \text{ est } A_n),$$

La valeur de vérité de la disjonction sur les univers du discours $X_1 \times X_2 \times \dots \times X_n$ est définie par:

$$V(p) = \max\{\mu_{A_1}(\mathcal{X}_1), \dots, \mu_{A_n}(\mathcal{X}_n)\}$$

Les implications entre propositions floues :

Règle 1 : Si $(\mathcal{X} \text{ est } A_1)$; alors $(\mathcal{Y} \text{ est } B_1)$,

Règle 2 : Si $(\mathcal{X} \text{ est } A_2)$; alors $(\mathcal{Y} \text{ est } B_2)$,

...

Règle n : Si $(\mathcal{X} \text{ est } A_n)$; alors $(\mathcal{Y} \text{ est } B_n)$.

Les implications seront présentées plus en détail dans le paragraphe suivant.

- Les combinaisons de conjonction, disjonction et implication de propositions floues élémentaires. Par exemple, «Si $(\mathcal{X}_1 \text{ est } A_{11})$ et $(\mathcal{X}_2 \text{ est } A_{12})$; alors $(\mathcal{Y} \text{ est } B_1)$ », etc.

II.5.3 Implications floues :

Dans la proposition floue $p : \text{«Si } (\mathcal{X} \text{ est } A) \text{ ; alors } (\mathcal{Y} \text{ est } B)\text{»}$, les propositions « \mathcal{X} est A » et « \mathcal{Y} est B » sont construites à partir des deux variables linguistiques qui sont a priori indépendantes. L'implication floue permet de définir une liaison entre la prémisse « \mathcal{X} est A » et la conclusion « \mathcal{Y} est B » de cette règle.

Définition 1.14 (Implications floues) :

Considérons $p : \ll X \text{ est } A \gg$, $q : \ll Y \text{ est } B \gg$ deux propositions floues Notons a, b les valeurs de vérité possibles de p et q respectivement.

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

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

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

$$A \Rightarrow B.$$

D'après la définition 1.14, les valeurs de vérité sont exprimées par

$$\begin{aligned} a &= \mu_A(x), \\ b &= \mu_B(y). \end{aligned}$$

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 $X \times Y$, 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 :

$$\mu_{R(x, y)} = I(\mu_A(x), \mu_B(x)).$$

Ainsi, la définition d'une implication floue peut se ramener à celle d'une fonction d'appartenance.

Les implications de Mamdani et de 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

Tableau II.1 : Implications floues les plus utilisées

II.5.4 Inférence floue :

Les méthodes d'inférence, utilisées habituellement en logique standard [6], peuvent être généralisées dans le cadre de la logique floue pour permettre de raisonner lorsque les règles ou les faits sont connus de façon imparfaite. La méthode d'inférence la plus connue est le modus ponens, qui permet de déduire une nouvelle connaissance en se basant sur la connaissance d'un seul fait et d'une seule règle.

II.6 Conclusion :

Nous présentons dans ce chapitre les concepts de la logique floue et nous décrivons les aspects méthodologiques nécessaires à la compréhension de cette technique en donnant des rappels sur les bases générales de la logique floue.

Chapitre III :

La commande floue

III. Introduction :

De façon générale, un système de commande 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 [14].

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 linguistiques 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. La figure (III.1) montre la structure générale d'un contrôleur floue.

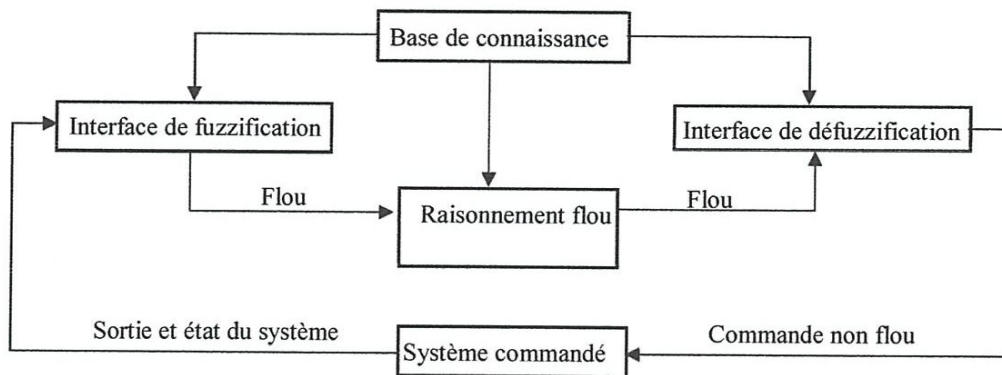


Figure III.1 : structure générale d'un contrôleur floue

III.1 Fuzzification :

La fuzzification est réalisée dans l'interface d'entrée du contrôleur flou. Durant cette phase, les informations issues du système sont tout d'abord normalisées. Ensuite, les données normalisées sont transformées en qualifications linguistiques, en utilisant des règles sémantiques définies par un expert.

Durant la phase de normalisation, chaque mesure issue du système est modifiée pour fournir une valeur appartenant à un univers du discours relativement simple. On peut choisir comme univers du discours un intervalle centré sur zéro : $[-c, +c]$. Si la mesure initiale x est comprise dans un autre intervalle $[a, b]$, la normalisation est souvent réalisée par transformation linéaire, selon :

$$y = \frac{2c}{b-a} \left[x - \frac{a+b}{2} \right] \quad (3.1)$$

L'univers du discours est ensuite représenté par une variable linguistique, qui comporte un nombre assez restreint de termes (en général trois, cinq ou sept) de façon à limiter le nombre de règles.

Enfin, les valeurs normalisées déduites de chacune des entrées sont transformées en qualifications linguistiques, en utilisant les variables linguistiques correspondantes.

Exemple 1 :

Une variable linguistique sur l'univers de discours $[-6, +6]$, peut être définie de la façon précisée dans le tableau :

Terme linguistique	Signification	Règle sémantique
NG	négatif grand	environ -6
NM	négatif moyen	environ -4
NP	négatif petit	environ -2
EZ	environ zéro	environ 0
PP	positif petit	environ +2
PM	positif moyen	environ +4
PG	positif grand	environ +6

Tableau III.1 : Exemple de variable linguistique.

La figure montre les fonctions d'appartenance associées aux différents termes linguistiques.

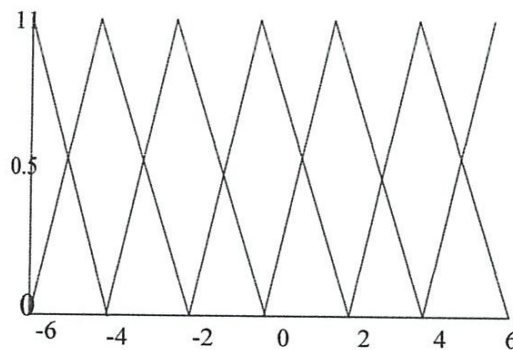


Figure III.2 : Univers du discours partitionné par les termes linguistique définis dans le tableau III.1

Dans cet exemple, les termes linguistiques utilisés sont ceux initialement proposés par Mamdani lorsque la variable linguistique en comporte sept. Les fonctions d'appartenance correspondant à des variables comportant trois et cinq termes sont représentées sur la figure(III.3) .

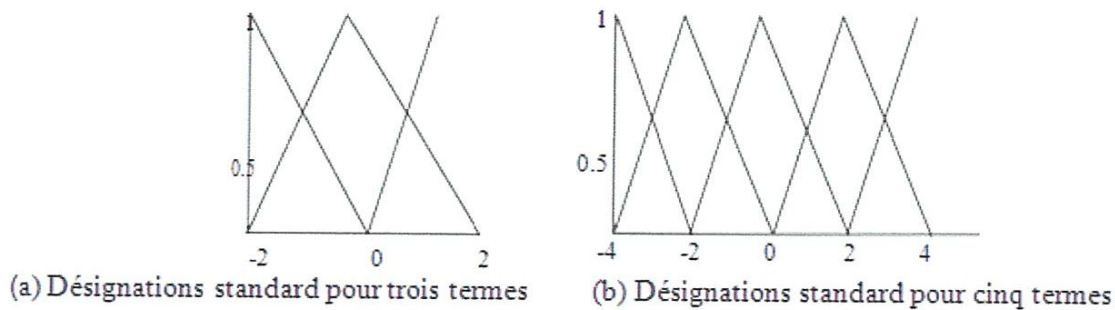


Figure III.3: Désignations standard et fonctions d'appartenance.

La fuzzification est une étape clé dans tout contrôleur flou. La variable linguistique, qui est définie par une expertise, doit respecter un certain nombre de critères afin d'être efficace :

- Chaque terme linguistique est un nombre flou, de noyau non nul et dont la fonction d'appartenance est convexe.
- Les fonctions d'appartenance doivent respecter l'ordre linguistique. Dans l'exemple 1.2, on ne peut pas modifier l'ordre des termes {froid, tiède, chaud} en {chaud, tiède, froid}.
- Les fonctions d'appartenance ne doivent pas trop se superposer. On tolère en général un chevauchement qui ne dépasse pas la mi-hauteur des termes linguistiques consécutifs.

III.2 Règles floues :

Les règles floues permettent de déduire des connaissances concernant l'état du système en fonction des qualifications linguistiques fournies par l'étape de fuzzification. Ces connaissances sont également des qualifications linguistiques.

Habituellement, les règles floues sont déduites des expériences acquises par les opérateurs ou les experts. Ces connaissances sont traduites en règles simples pouvant être utilisées dans un processus d'inférence floue. Par exemple, si un expert exprime la règle «si la température de l'eau est chaude, il faut ajouter de l'eau froide», le système utilisera une règle du genre «si p alors q ».

Aujourd'hui, il est cependant possible de constituer une base de règles floues grâce à des méthodes d'apprentissage, sans avoir nécessairement besoin d'un expert humain.

Cette stratégie sera décrite plus en détail dans la suite de cette thèse.

III.3 Inférences floues :

Dans la section II.5.5 Nous avons constaté qu'une inférence floue n'est ni plus ni moins qu'une relation floue définie entre deux sous-ensembles. La définition de la relation peut théoriquement faire intervenir n'importe quel opérateur de combinaison. Pourtant, on utilise souvent les inférences floues définies par Mamdani et Sugeno.

III.3.1 Inférence floue de Mamdani :

Supposons que la base de connaissances est constituée de n règles d'inférence contenant chacune m prémisses et une conclusion. Le fait est également constitué de m propositions floues. Le processus d'inférence peut être décrit par le schéma suivant [7] :

$$\begin{array}{l}
 \text{Règle 1:} \quad \text{Si } (\mathcal{X}_1 \text{ est } A_{11}) \text{ et } \dots \text{ et } (\mathcal{X}_m \text{ est } A_{1m}) \quad ; \text{ alors } (\mathcal{Y} \text{ est } B_1) \\
 \text{Règle 2:} \quad \text{Si } (\mathcal{X}_1 \text{ est } A_{21}) \text{ et } \dots \text{ et } (\mathcal{X}_m \text{ est } A_{2m}) \quad ; \text{ alors } (\mathcal{Y} \text{ est } B_2) \\
 \dots \\
 \text{Règle n:} \quad \text{Si } (\mathcal{X}_1 \text{ est } A_{n1}) \text{ et } \dots \text{ et } (\mathcal{X}_m \text{ est } A_{nm}) \quad ; \text{ alors } (\mathcal{Y} \text{ est } B_n) \\
 \hline
 \text{Fait :} \quad \frac{(\mathcal{X}_1 \text{ est } A'_1) \text{ et } \dots \text{ et } (\mathcal{X}_m \text{ est } A'_m)}{\quad} \\
 \hline
 \text{Conclusion :} \quad (\mathcal{Y} \text{ est } B')
 \end{array}$$

dans lequel $\mathcal{X}_1, \dots, \mathcal{X}_m$ sont des éléments des universs du discours X_1, \dots, X_m et $A_{ji}, (j = 1, \dots, m)$, A'_i sont des quantités floues sur l'univers du discours X_i , et $B_j, (j = 1, \dots, m)$, B' sont également des quantités floues sur l'univers du discours Y .

Afin de définir une seule prémisse pour une règle i , les propositions « \mathcal{X} est A_{ij} », $(j = 1, \dots, m)$, sont combinées par l'opérateur minimum. La fonction d'appartenance de cette prémisse unique est donc donnée par :

D'après Mamdani, l'implication $A \Rightarrow B$ est également construite par un opérateur de minimum, selon :

$$(A \Rightarrow B) = (A \wedge B).$$

On peut ainsi déterminer simplement la fonction d'appartenance de la règle R_i :

$$\mu_{R_i}(x_1, \dots, x_m, y) = \mu_{R_i}(x_1, \dots, x_m) \wedge \mu_{B_i}(y)$$

La figure III.4 montre un synoptique de ce processus d'inférence. Dans cet exemple, au niveau des conditions, les opérateurs logiques standard «et» et «ou» sont remplacés respectivement par les opérateurs de minimum et de maximum. La conclusion de chaque règle, introduite par «alors» est également calculée par l'opérateur minimum. C'est pourquoi on appelle également l'inférence de Mamdani inférence max-min.[15]

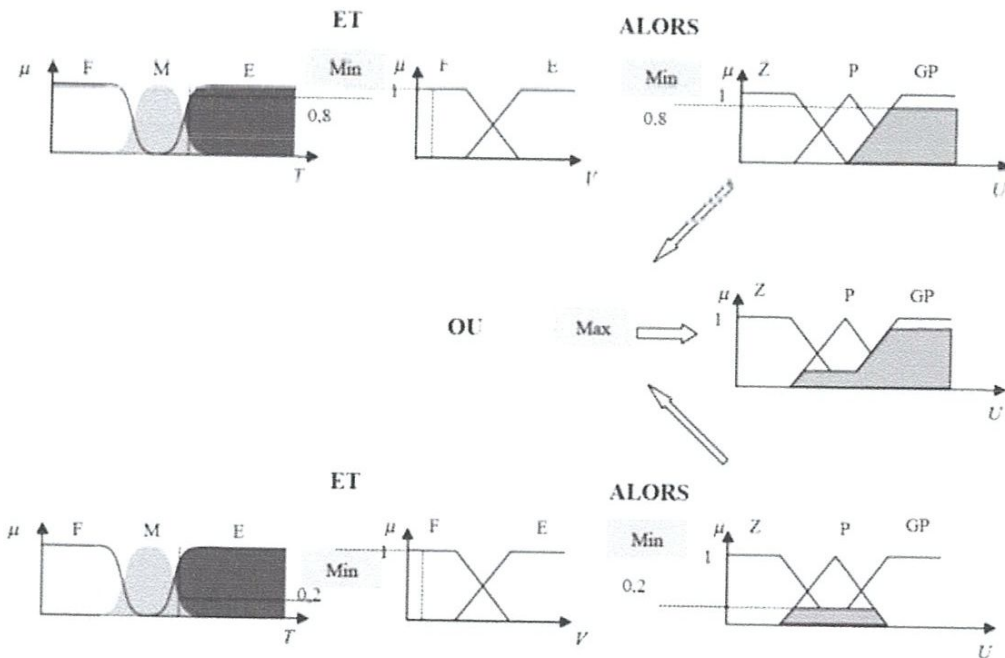


Figure III.4 : Exemple d'inférence max-min.

L'inférence max-min est une méthode relativement simple largement utilisée dans les applications industrielles. Elle est parfois remplacée par l'inférence max-prod, ou inférence de Larsen, qui permet d'améliorer la qualité du résultat. Pour cette inférence, plutôt que d'utiliser l'opérateur minimum pour définir la conclusion d'une règle, on utilise le produit.

La figure (III.5) présente un synoptique du traitement de l'exemple de la figure (III.4) par l'inférence max-prod.

Exemple 2:

Soit un système à commander, dans lequel on analyse deux valeurs : l'erreur E et le taux d'erreur ΔE . Le contrôleur flou doit fournir la commande S . Les valeurs d'entrée du contrôleur sont transformées en termes linguistiques dont les fonctions d'appartenance sont

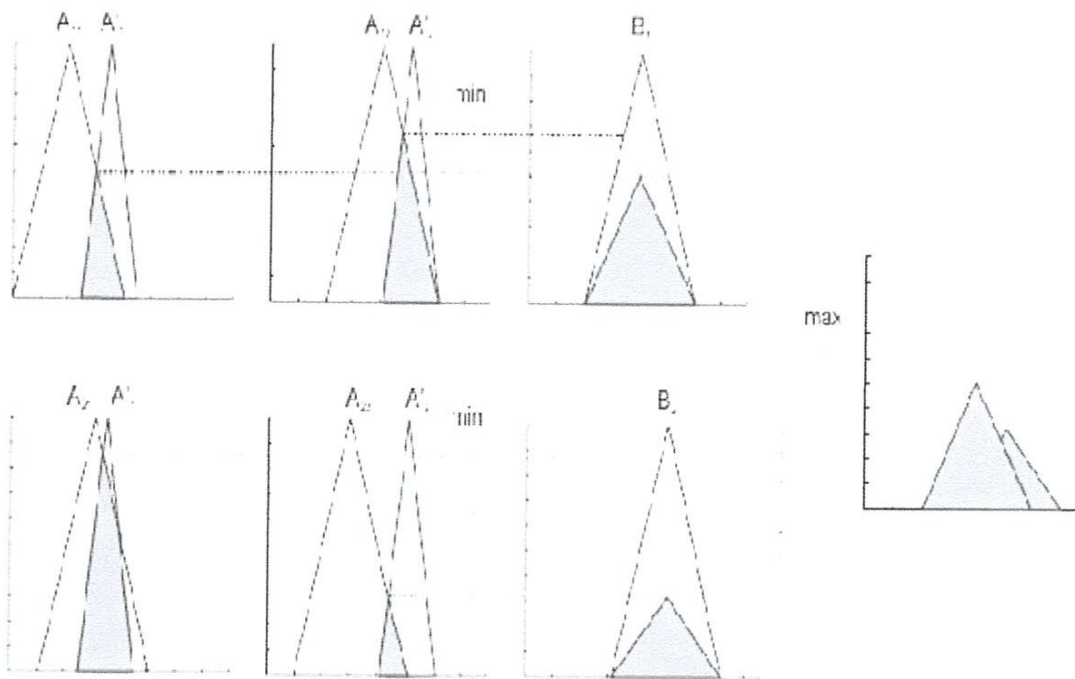


Figure III.5 : Exemple d'inférence max-prod.

III.3.2 Inférence floue de Sugeno :

Sugeno a proposé une méthode d'inférence floue qui garantit la continuité de la sortie [11]. Cette méthode d'inférence s'avère très efficace dans des applications faisant intervenir à la fois des techniques linéaires, d'optimisation et adaptatives. Dans l'inférence de Sugeno, les règles floues sont exprimées de la façon suivante :

Règle i : Si (x_1 est A_{i1}) et \dots et (x_m est A_{im}) ; alors $y = f_i(x_1, \dots, x_m)$

dans laquelle x_1, \dots, x_m et y sont des éléments des universs du discours X_1, \dots, X_m et A_{i1}, \dots, A_{im} sont des termes linguistiques sur ces mêmes universs du discours. y est une fonction de x_1, \dots, x_m .

Le problème consiste à déterminer les paramètres de la fonction, ce qui est possible en utilisant une méthode d'optimisation [16]. Une méthode utilisant un réseau de neurones comme système d'optimisation a été décrite dans [17]. Par rapport à l'inférence de Sugeno, celle de Mamdani est plus intuitive, plus générale et elle s'adapte particulièrement bien à l'utilisation de connaissances issues d'une expertise humaine.

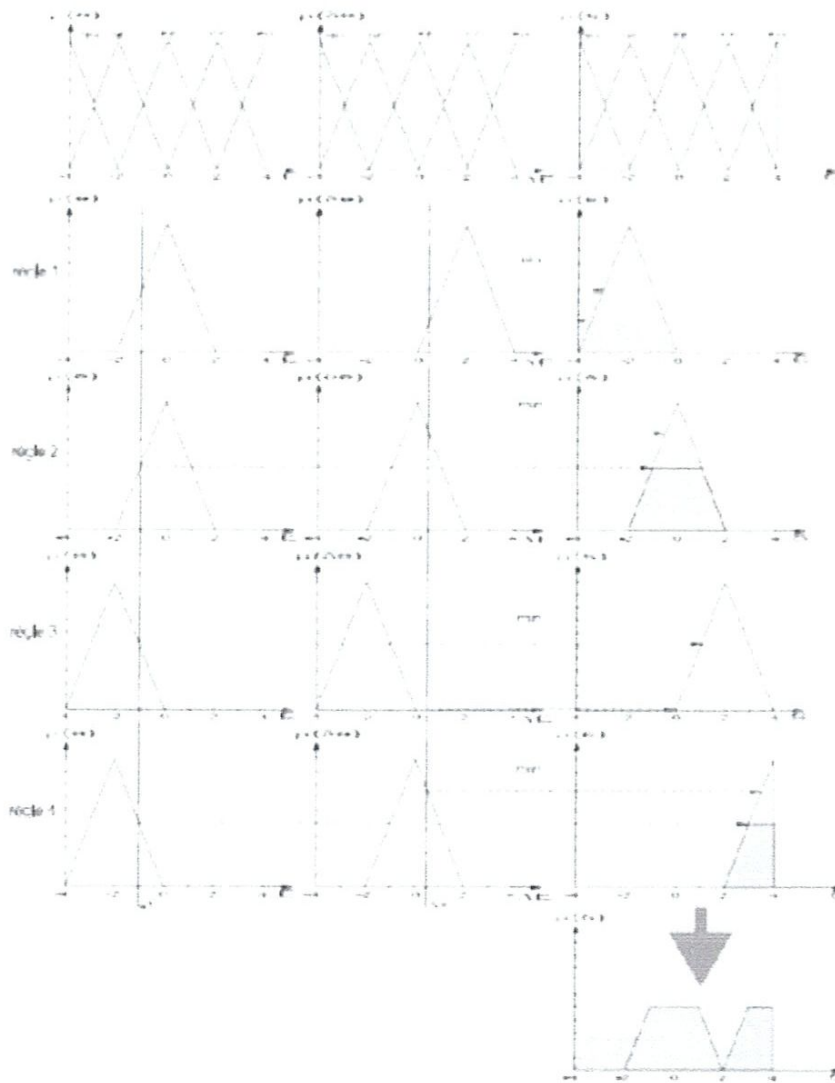


Figure III.6 : Exemple d'inférence de Mamdani.

III.4 Défuzzification :

Comme nous avons vu dans la section précédente, les méthodes d'inférence fournissent un résultat qui est une fonction d'appartenance. Or, la sortie du contrôleur est en général une grandeur continue, prenant sa valeur dans un intervalle. La défuzzification est le traitement qui permet de définir une correspondance entre le résultat de l'inférence et la grandeur continue fournie en sortie.[11]

III.4.1 Défuzzification par centre de gravité :

La défuzzification par centre de gravité consiste à calculer l'abscisse du centre de gravité de la fonction d'appartenance selon :

$$y_{cg} = \frac{\int_y y \cdot \mu_{B_{res}}(y) dy}{\int_y \mu_{B_{res}}(y) dy} \quad (1.16)$$

Le temps nécessaire au traitement est directement proportionnel au nombre de points retenus pour le calcul de la moyenne. Selon les contraintes fixées par l'application, il y a un compromis à réaliser entre la précision souhaitée et le temps de calcul disponible.

III.4.2 Défuzzification par centre 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. Cette valeur est fournie par l'expression :

$$y_{cm} = \frac{\inf M + \sup M}{2} \quad (1.17)$$

III.4.3 Défuzzification par valeur maximum :

Cette méthode ne s'utilise que dans le cas discret. On choisit comme sortie y_m l'abscisse de la valeur maximale de la fonction d'appartenance résultante $\mu_{B_{res}}(y)$. Lorsque $\mu_{B_{res}}(y)$ est échantillonnée, on prend la moyenne des abscisses du maximum :

$$Y_m = \frac{\sum_{Y_i \in M} Y_i}{|M|} \quad (1.18)$$

III.5 Avantages et Inconvénients de Réglage par Logique Floue :

III.5.1 Avantages :

- Pas besoin de modèle mathématique.
- Possibilité d'implémenter des connaissances linguistiques.
- Maîtrise de systèmes à régler avec un comportement complexe.
- Disponibilité des systèmes de développement efficaces, soit pour microprocesseur ou PC « solution logiciel », soit pour les circuits intégrés (processeurs dédiés, fuzzy processors), « solution matérielle ».

III.5.2 Inconvénients :

- Manque de directives précises pour la conception d'un réglage.
- Les règles doivent être disponibles.
- Approche artisanale et non systématique.

Aucune méthode formelle pour l'ajustement

III.6 Conclusion :

Nous avons vu dans ce chapitre l'intérêt de la logique floue dans le domaine du contrôle de processus. Cette approche permet de tenir compte à la fois des connaissances d'un expert humain et de l'incertitude et de l'imprécision des données traitées par le contrôleur.

Les variables linguistiques permettent de traiter ces deux informations initialement très différentes à l'aide d'un formalisme unique.

Pourtant, la conception d'un contrôleur flou n'est pas toujours chose aisée. Lorsqu'on utilise un contrôleur de type standard (par exemple un PID), on dispose de nombreux outils de synthèse permettant de choisir au mieux les paramètres du régulateur en fonction de la structure ou du modèle du système à commander. Malheureusement, la panoplie d'outils disponibles est beaucoup plus limitée dans le cas des contrôleurs flous.

En fait, le fonctionnement d'un contrôleur flou dépend d'un nombre très important de paramètres (fonctions d'appartenance, règles floues, règles d'inférence, défuzzification) qu'il faut régler lors de la conception.

Chapitre IV :

Application

IV.1 Introduction

L'être humain résout souvent des problèmes complexes à l'aide de données approximatives : la précision des données est souvent inutile ; par exemple pour choisir un appartement, il pourra prendre en compte la surface, la proximité de commerces, la distance du lieu de travail, le loyer, sans pour autant avoir besoin d'une valeur très précise de chacune de ces données ; la logique floue est bien connue des automaticiens pour ses applications dans le contrôle-commande de procédés, appelé alors couramment « contrôle flou ». Tout comme un contrôleur (ou correcteur) classique, le contrôleur flou s'insère dans la boucle de régulation et calcule la commande à appliquer au procédé suivant une ou plusieurs consignes et une ou plusieurs mesures effectuées sur celui-ci.

On présente ici le cas de réglage de la température d'une serre agricole. Dans ce but, on mesure la température avec un capteur qui fournit la valeur de la grandeur à régler [18].



Figure IV.1 : Une serre agricole.

On suppose que le système est linéaire et ayant une fonction de transfert :

$$H(s) = \frac{1}{s+2} e^{-L}$$

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 Matlab/Simulink [19].

Pour la régulation de ce système, on propose deux solutions:

- 1^{er} cas : Régulation avec un correcteur PID.
- 2^{ème} cas : Régulateur avec un correcteur floue.

IV.2 Régulation avec un Correcteur PID

❖ Réglage (calcul des actions PID)

✓ On utilise la Méthode de Ziegler Nichols en chaîne fermée :

Cette méthode, établie vers 1942, est la plus connue des méthodes pratiques de réglage des boucles de régulation.

L'avantage de cette méthode est qu'il n'y a pas besoin de connaître la fonction de transfert du procédé et que le réglage se fait directement sur le site en chaîne fermée. Après avoir porté la mesure près du point de consigne désiré (manuellement ou automatiquement), le régulateur est réglé en action proportionnelle seule. Le gain KR, est alors augmenté progressivement jusqu'à obtention du pompage sans que l'organe réglant ne soit jamais en saturation. Le gain critique KRC du régulateur est le plus petit gain qui permet l'entretien des oscillations. La période d'oscillation est mesurée sur l'enregistrement (figure IV.2).

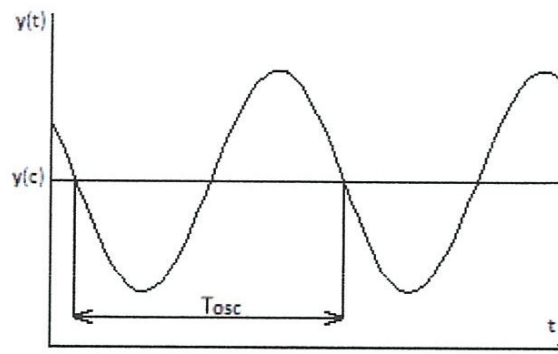


Figure IV.2 : Mise en oscillation juste entretenues d'un système

Ziegler et Nichols proposent alors les valeurs de réglage du tableau suivant

	P	PI	PI Parallèle	PID Série	PID Parallèle	PID Mixte
Krc	$\frac{Krc}{2}$	$\frac{Krc}{2.2}$	$\frac{Krc}{2.2}$	$\frac{Krc}{3.3}$	$\frac{Krc}{1.7}$	$\frac{Krc}{1.7}$
Ti	Ti max ou annulée	$\frac{Tosc}{1.2}$	$\frac{2.Tosc}{Krc}$	$\frac{Tosc}{4}$	$\frac{0.85 Tocs}{Krc}$	$\frac{Tocs}{2}$
Td	0	0	0	$\frac{Tocs}{4}$	$\frac{Krc Tocs}{13.3}$	$\frac{Tocs}{8}$

Tableau IV.1 : Réglages préconisés par Ziegler et Nichols.

On règle le PID selon les paramètres suivants :

PID (x1=1.7880; x2=0.7676; x3=0.6274)

- **Schéma Simulink**

Pour la simulation de notre procédé, on va utiliser pour chaque régulateur, soit PID ou logique Floue un schéma simulink particulier.

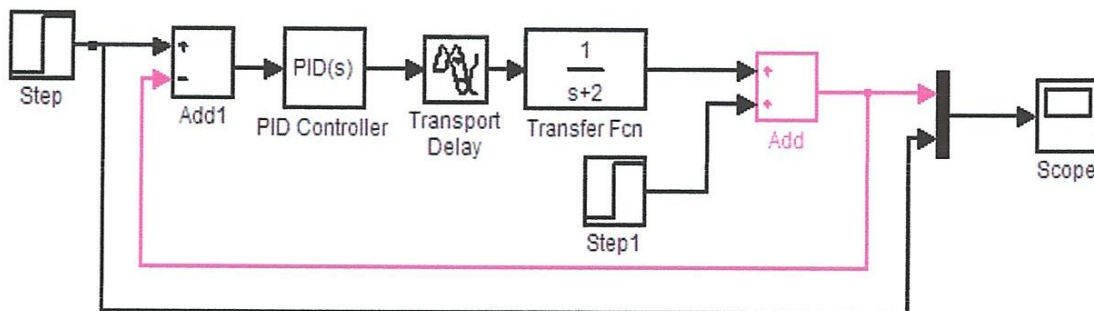


Figure IV.3 : Régulation avec un correcteur PID

On trace à la même figure l'entrée et la sortie (la réponse) de ce système :

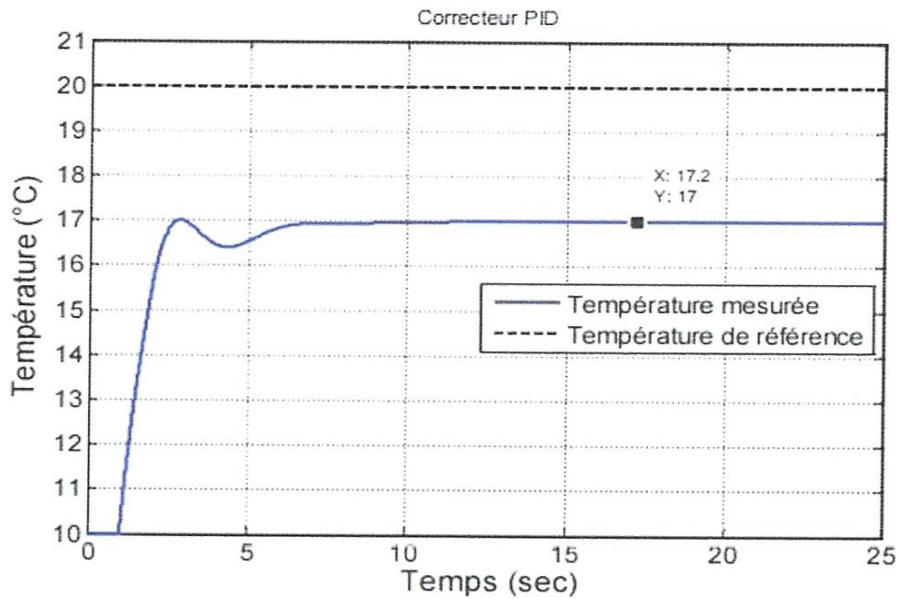


Figure IV.4: Réponse du système avec un correcteur PID

IV.3 Régulation par logique floue

On considère alors le même système précédant mais ici on remplace le contrôleur PID classique par un contrôleur basé sur la logique floue « FuzzyLogic Controller »,

IV.3.1 Régulation à une seule entrée :

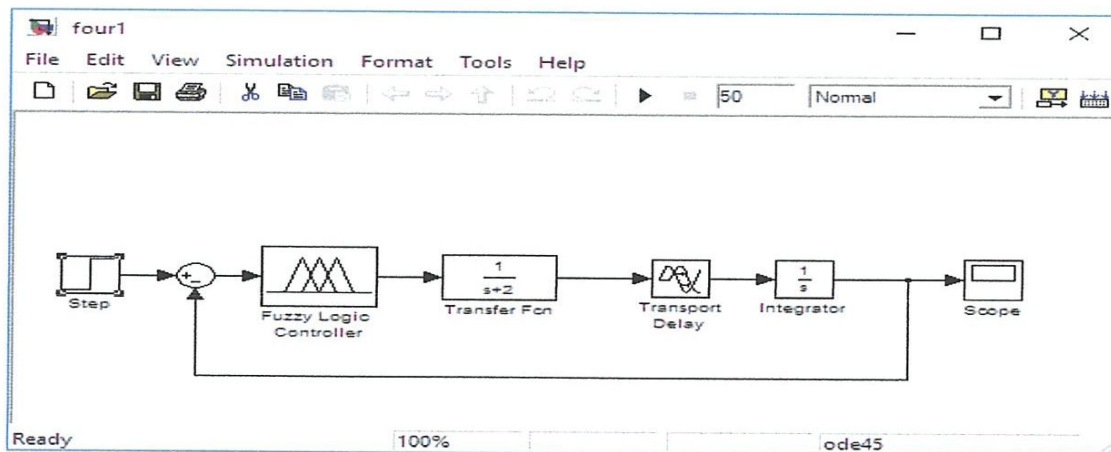


Figure IV.5 : Schéma Simulink d'un correcteur flou a une seule entrée

Pour construire notre régulateur flou (FuzzyLogic Controller), on commence tout d'abord par définir l'entrée et la sortie, On prendra pour entrée l'erreur, c'est l'écart de température et

pour sortie **la commande** (On commence en tapant fuzzy à la ligne de commande, on obtient alors le fis Editor de Matlab).

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 fonctions d'appartenances utilisées sont de type **trimf** (triangulaires). Une fois le type de fonction d'appartenance choisi ainsi que la plage de variation on appellera chaque partie de la fonction d'appartenance par un nom significatif 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 milieu sont assignés à une **commande** appelée **rien**.

Les règles d'inférences seront alors définies comme suit :

- Si c'est **chaud** alors **refroidir**.
- Si c'est **Ok** alors **rien** faire.
- Si c'est **froid** alors **réchauffer**.

Le système d'inférence flou sera de type Mamdani et la défuzzification se fera avec la méthode centroïde.

On simule le système et on trouve une erreur flagrante sur la réponse, pour y remédier on a changé les plages de variation des entrée et de la sortie sur notre régulateur;[20] on a pris alors:

- [-10,14] et après changé [-8.5 14] plage de variation de l'erreur (entrée).
- [-30, 37] plage de variation de la commande (sortie).

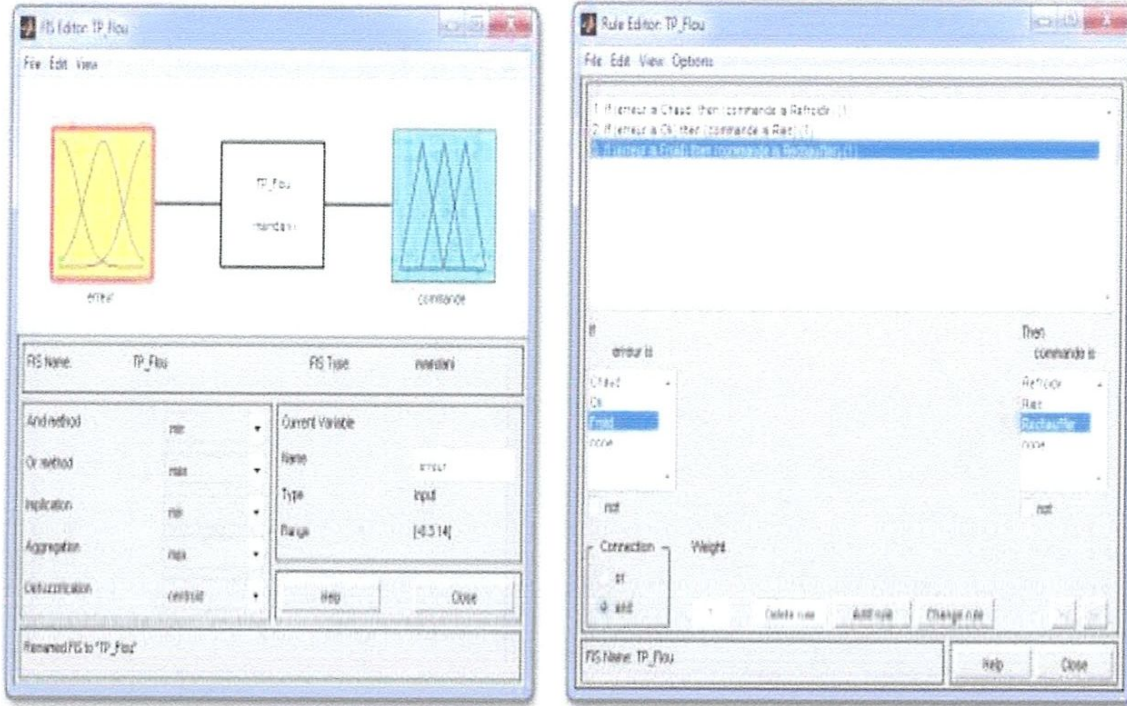


Figure IV.6 : Interface de la logique floue sous Matlab

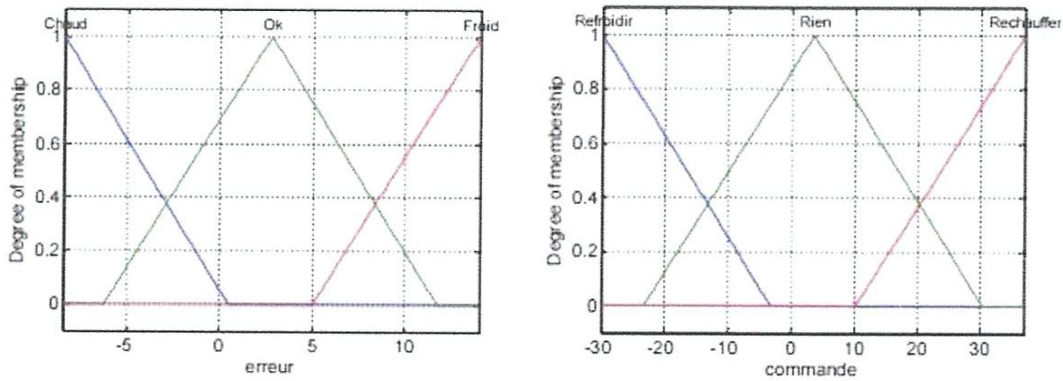


Figure IV.7: Fonctions d'appartenance de l'entrée (erreur) et de la sortie (commande)

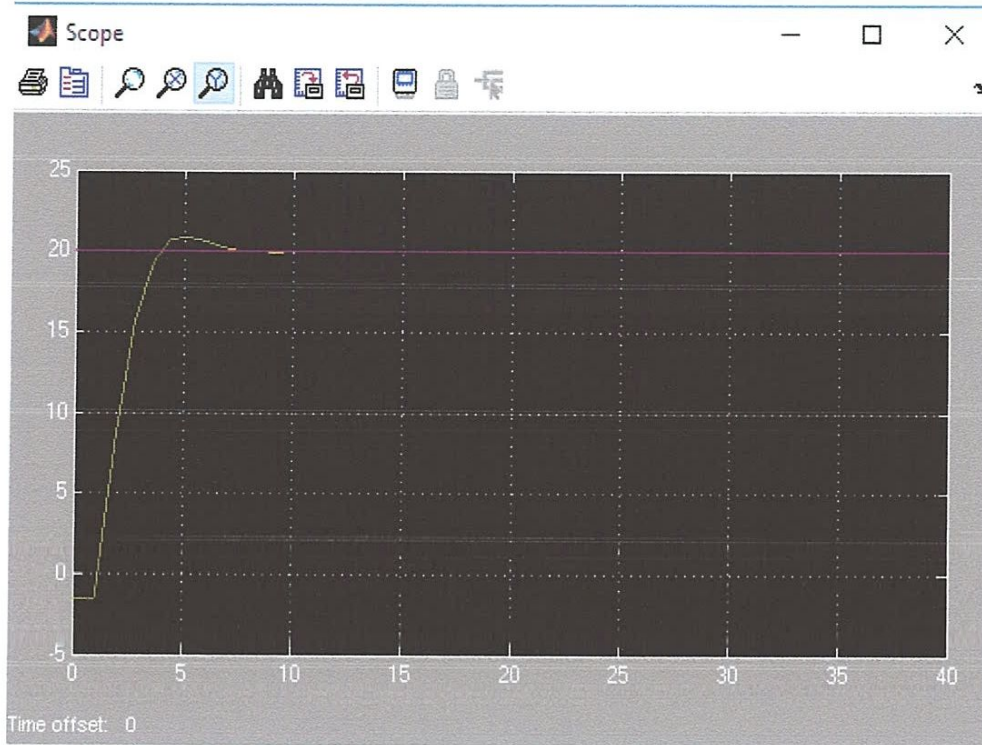


Figure IV.8 : Réponse du système avec un correcteur flou a une seule entrée

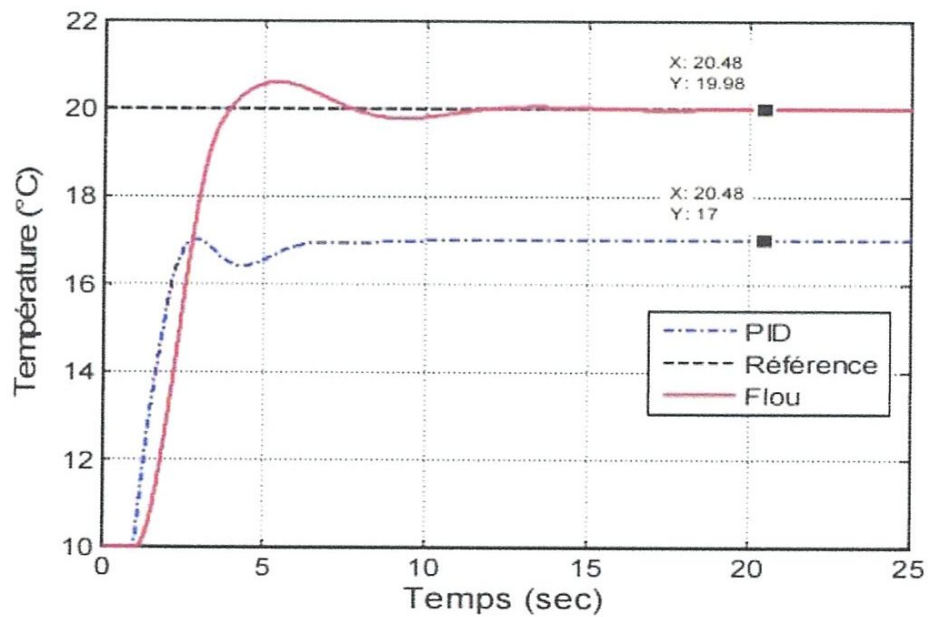


Figure IV.9: Réponses du système avec un correcteur PID et flou

- D'après la figure (IV.9) , on remarque que la régulation par un correcteur flou a les meilleures performances que le correcteur PID.

IV.3.2 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, c'est la dérivée de l'erreur

- Ensembles flous :

On va exploiter 3 variables, l'erreur (notée e), entrée du régulateur, mais aussi la variation de l'erreur (notée Δe) et la sortie (notée s).

L'entrée e a été partitionnée en 3 valeurs linguistiques (chaud, Ok et froid) ainsi que e . Elles peuvent être positives ou négatives, situées sur un intervalle de $[-10, 14]$ et $[-15, 15]$ respectivement pour e et Δe [20].

La sortie s a été partitionnée en 5 valeurs linguistiques ; elle pourra être positive ou négative.

Les figures (IV.11, IV.12) présentent les fonctions d'appartenance des entrées.

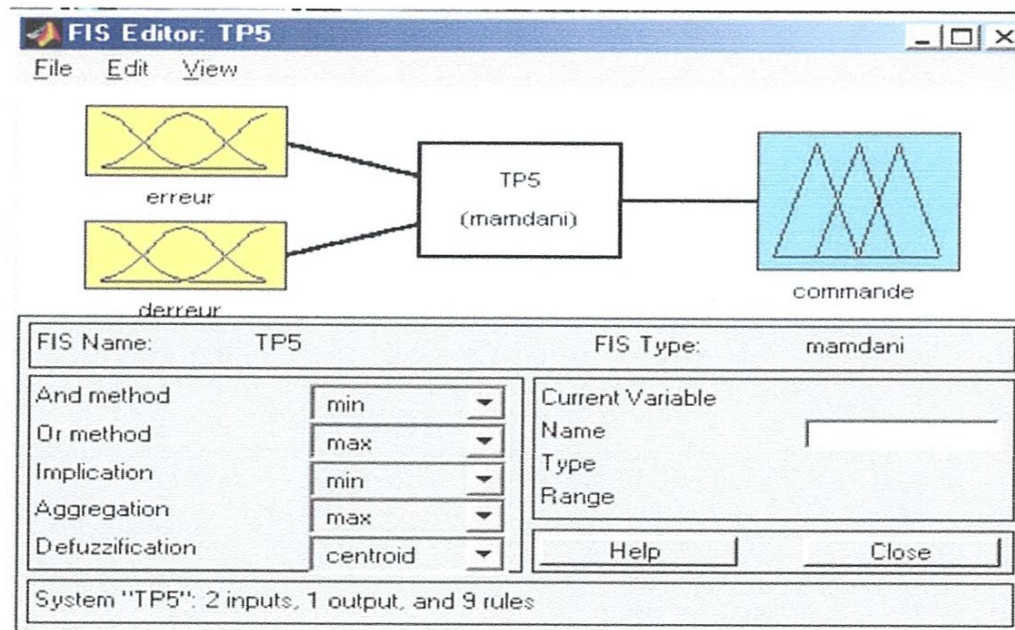


Figure IV.10 : Interface de la logique floue sous Matlab

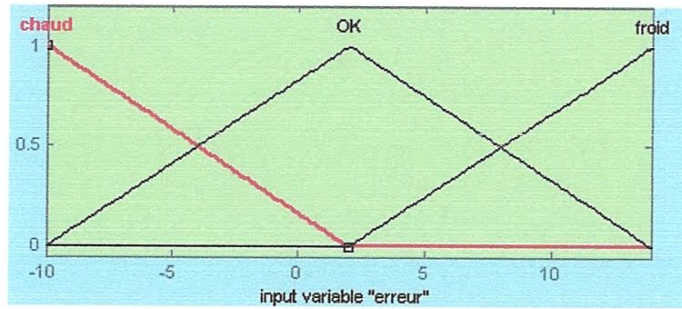


Figure IV.11: Fonctions d'appartenance de l'entrée (erreur)

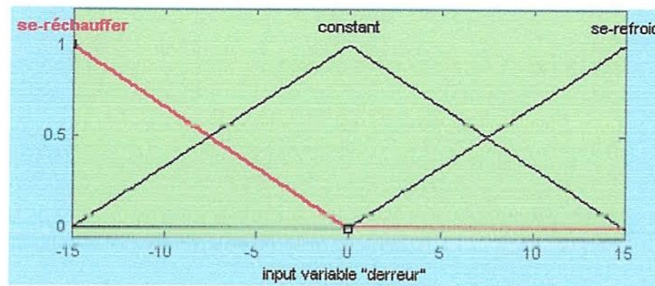


Figure IV.12: Fonctions d'appartenance de l'entrée (derreur)

La figure (IV.13) présente les fonctions d'appartenance de sortie. Cette fonction d'appartenance a été divisée en 5 parties :

- Refroidir beaucoup : RB
- Refroidir : R
- ok
- Réchauffer : C
- Réchauffer beaucoup : CB

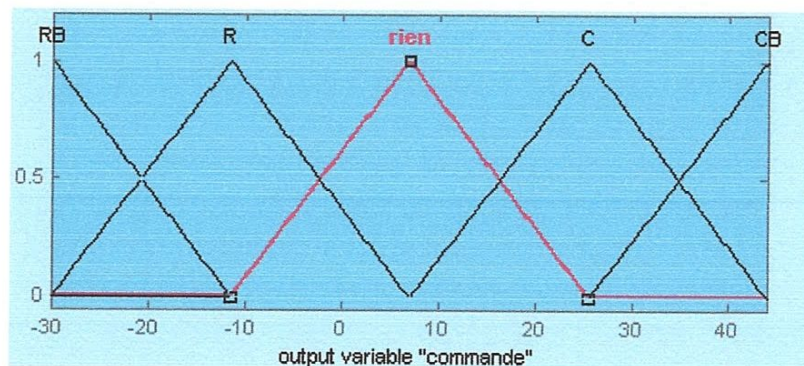


Figure IV.13: Fonctions d'appartenance de sortie (commande)

- Les règles Floues :

Les règles floues utilisées pour deux entrées sont :

- ✓ R1 : **SI (e) est chaud ET (Δe) est se-refroid ALORS (s) est R**
- ✓ R2 : **SI (e) est chaud ET (Δe) est constant ALORS (s) est R**
- ✓ R3 : **SI (e) est chaud ET (Δe) est se-réchauffer ALORS (s) est RB**
- ✓ R4 : **SI (e) est OK ET (Δe) est se-refroid ALORS (s) est C**
- ✓ R5 : **SI (e) est OK ET (Δe) est constant ALORS (s) est rien**
- ✓ R6 : **SI (e) est OK ET (Δe) est se-réchauffer ALORS (s) est R**
- ✓ R7 : **SI (e) est froid ET (Δe) est se-refroid ALORS (s) est CB**
- ✓ R8 : **SI (e) est froid ET (Δe) est constant ALORS (s) est C**
- ✓ R9 : **SI (e) est froid ET (Δe) est se-réchauffer ALORS (s) est C**

On résumera les règles d'inférence au tableau suivant :

Erreur Variation d'erreur	Chaud	Ok	Froid
Se refroidit	R	C	CB
Constant	R	Rien	C
Se réchauffe	Rb	R	C

Tableau IV.2 : Les règles floues

Pour le Régulateur Floue :

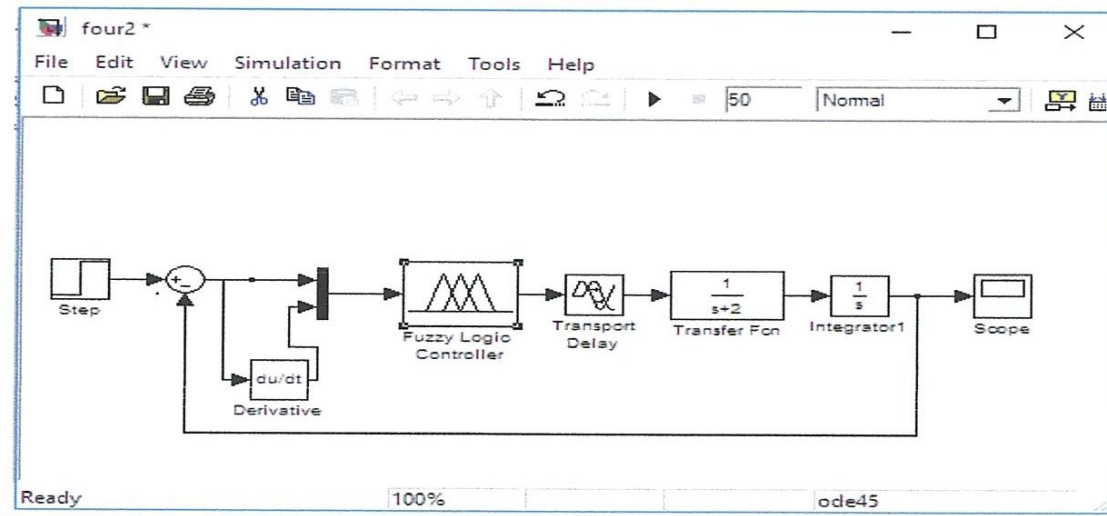


Figure IV.14 : Schéma Simulink d'un correcteur flou a deux entrées

Point de fonctionnement nominal : température de serre 10 °C.

Pour des échelons de consigne et de perturbation constants, La température est fixée à 20°C ,
On effectue une série de changement de retard (Figures IV.15, 16, 17, 18, 19) et on simule le comportement du procédé par les deux commandes PID et logique Floue.

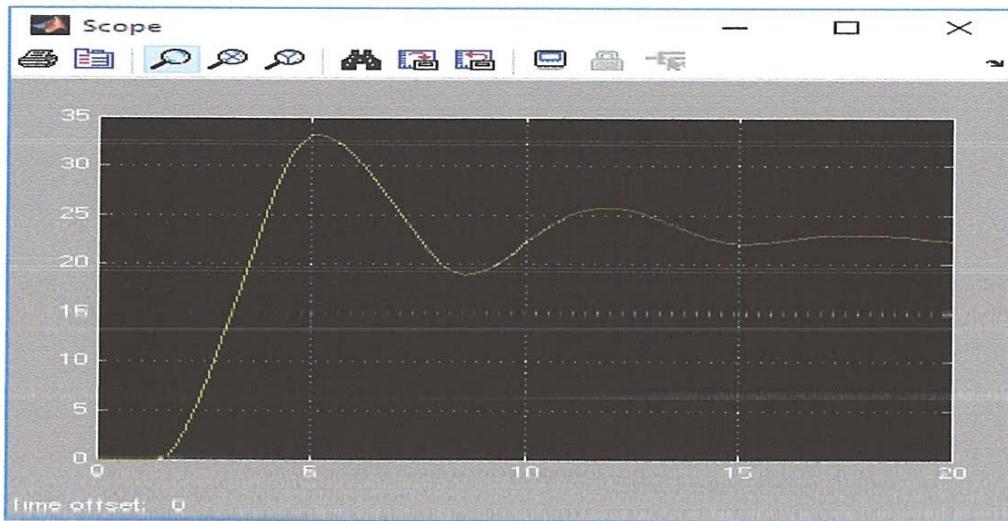


Figure IV.15 : Commande PID pour un retard= 1.5s

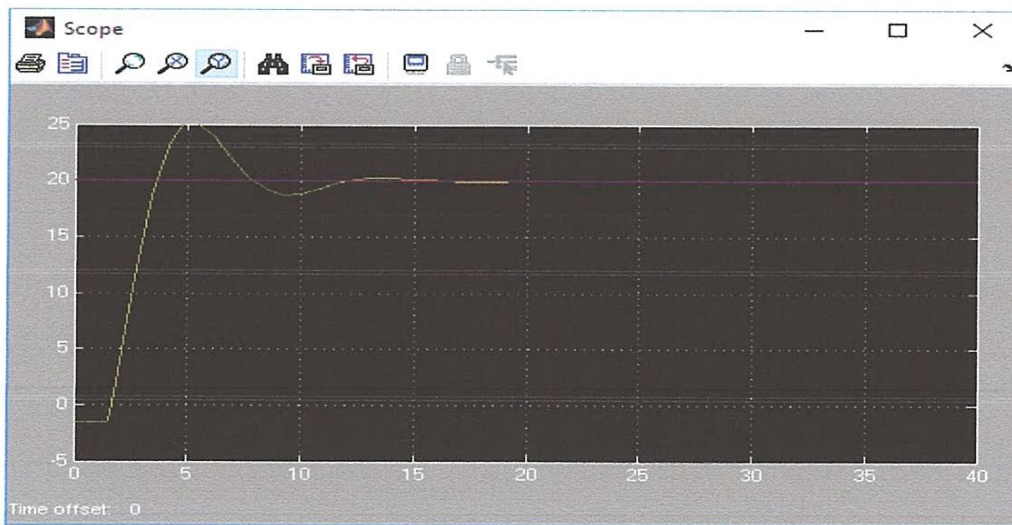


Figure IV.16 : Commande floue pour un retard= 1.5s

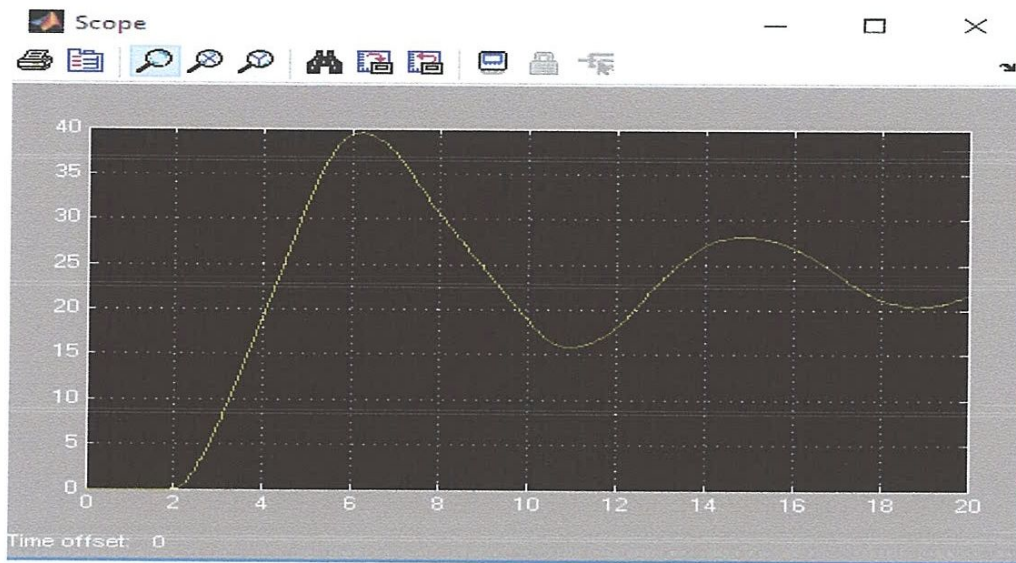


Figure IV.17 : Commande PID pour un retard= 2 s

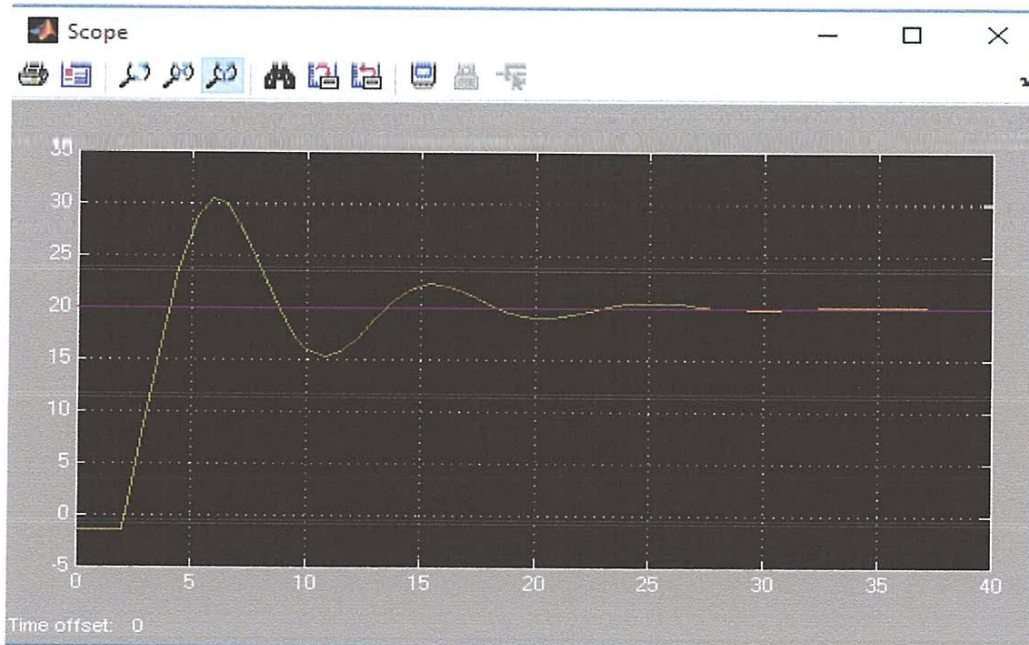


Figure IV.18 : Commande floue pour un retard= 2s

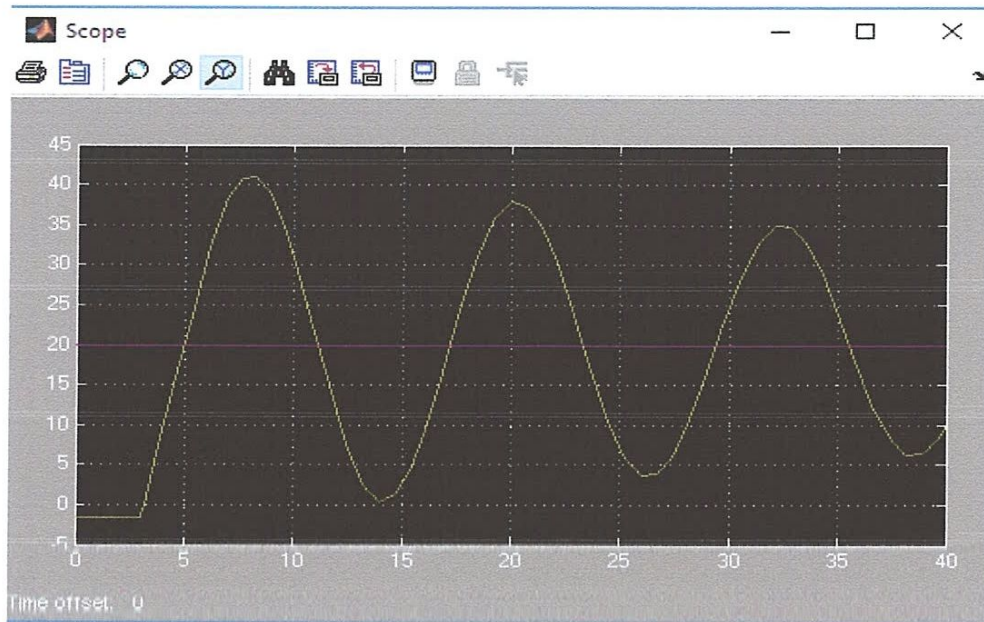


Figure IV.19 : Commande PID pour un retard= 3s

➤ **Interprétation des résultats :**

Si on remarque les critères de performances (précision, temps de réponse à 5% et dépassement). On constate que les deux régulateurs donnent des résultats acceptables pour les petits et moyens changements de retard, mais pour des grands retards le PID perd ses performances, en particulier en terme de dépassement ou temps de réponse. Par contre le régulateur flou reste robuste.

IV.4 Autre exemple explicatif :

Pour mieux illustrer le concept d'un contrôleur flou, on comparera la performance d'un contrôleur PID classique avec le contrôleur flou. Le système sous étude est un système d'ordre trois donné par la fonction de transfert suivante [21] :

$$G(s) = \frac{3}{(s+1)(s+2)(s+3)}$$

En utilisant les méthodes classiques (ziegler nickols), on trouve que le contrôleur PID idéal a les valeurs suivantes:

Kp	12
Ki	12
Kd	3

On utilisera la réponse à une entrée échelon du système avec PID pour comparer les performances. Le système avec contrôleur flou et contrôleur PID devrait être le suivant :

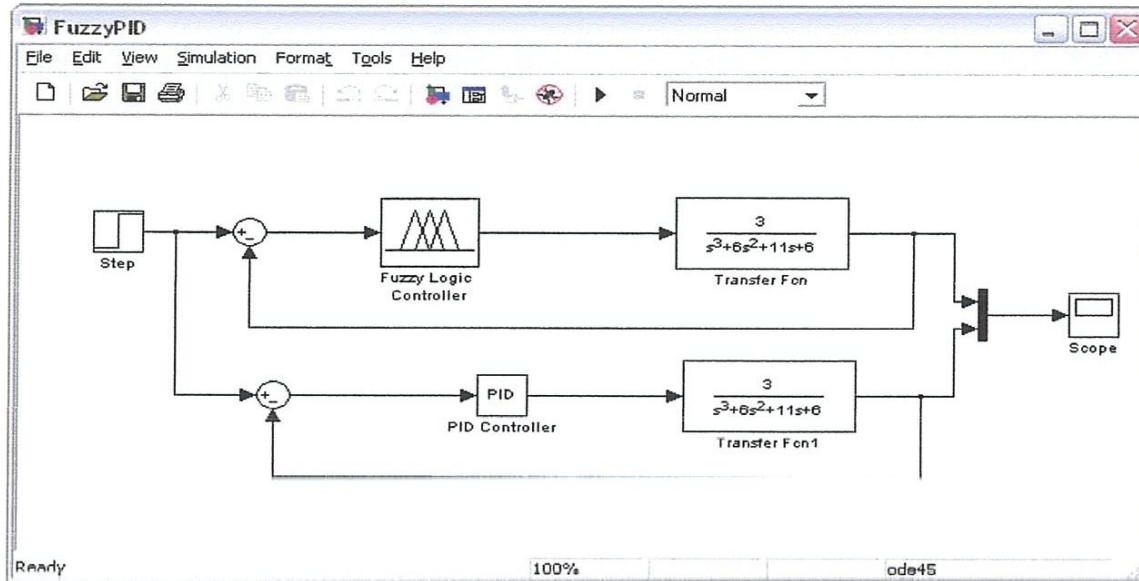


Figure IV.20 : Schéma Simulink d'un correcteur flou et PID.

Pour le système simple sous étude, on utilise que l'erreur comme variable de contrôle. On a donc un système très simple, avec une entrée, une sortie.

Il faut utiliser un peu de jugement dans le design du contrôleur flou. Il ne faut pas simplement créer des fonctions d'appartenance de toute façon. Il faut quand même étudier le système en détail. Une chose importante à remarquer est la valeur finale du système en boucle ouverte. Ceci déterminera la sortie du contrôleur quand l'erreur est nulle. Pour le système sous étude, la valeur finale due à une entrée échelon unitaire est :

$$y(\infty) = \lim_{s \rightarrow 0} sR(s)G_0(s) = \lim_{s \rightarrow 0} s \cdot \frac{1}{s} \cdot \frac{3}{(s+1)(s+2)(s+3)} = \frac{1}{2}$$

$$Kc = \frac{1}{y(\infty)} = 2$$

C'est-à-dire que le gain du contrôleur flou lorsque l'erreur (l'entrée) est 0, est 2. Il faut aussi bien choisir les limites du système (limites de l'entrée et de la sortie). Dans ce cas-ci, l'entrée utilisée est une entrée échelon unitaire. On choisit donc de développer les fonctions d'appartenance du contrôleur flou pour l'entrée entre -1 et +1.

On a choisi que 3 valeurs linguistiques possibles, soit *Négative*, *Zéro* et *Positive*. Il est généralement plus facile de commencer avec un nombre restreint de valeurs linguistiques possibles, pour simplifier le design initial. Par après, si la précision demandée est plus grande, on peut ajouter des valeurs linguistiques.

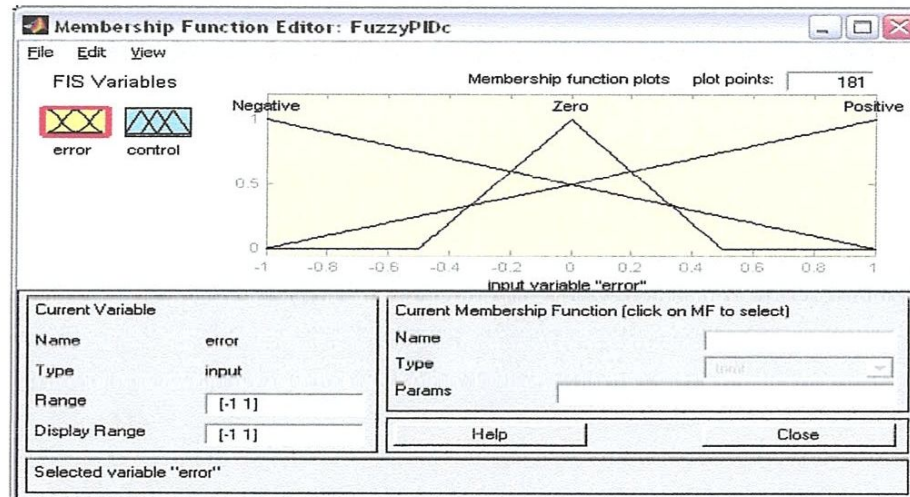


Figure IV.21: Fonctions d'appartenance de l'entrée (erreur)

On choisit aussi 3 valeurs linguistiques pour la sortie, soit *Négative*, *Zéro* et *Positive*.

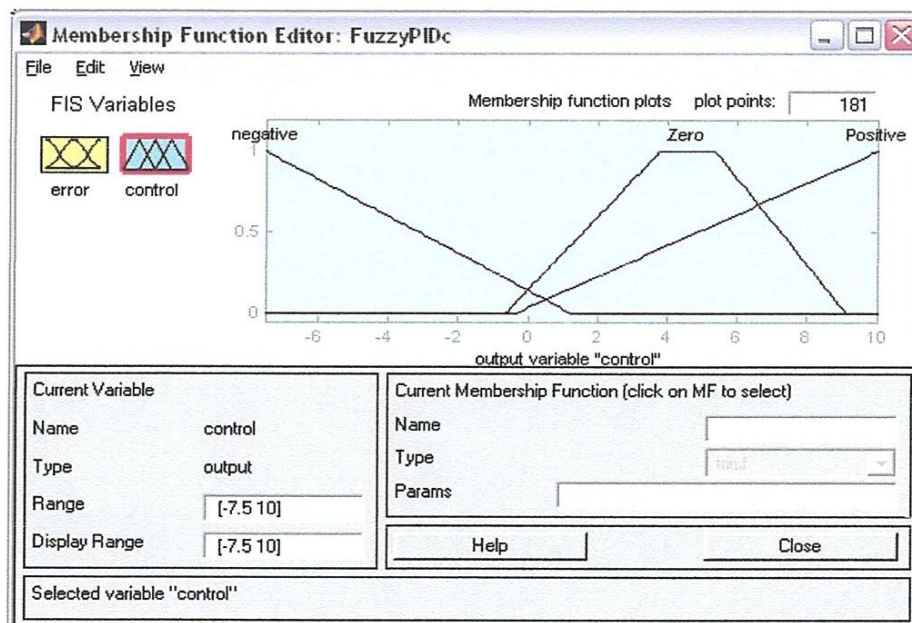


Figure IV.22: Fonctions d'appartenance de sortie (commande)

Remarquez que l'intervalle de la sortie est de -7.5 à $+10$. Le signal de sortie peut être bien plus grand que le signal d'entrée afin de rapidement corriger l'erreur. Cependant, si le signal de contrôle est trop élevé, il y aura un plus grand dépassement.

Les règles utilisées sont:

1. IF error is Negative THEN control is Negative
2. IF error is Zero THEN control is Zero
3. IF error is Positive THEN control is Positive

Il ne reste qu'à simuler le système et comparer avec le PID :

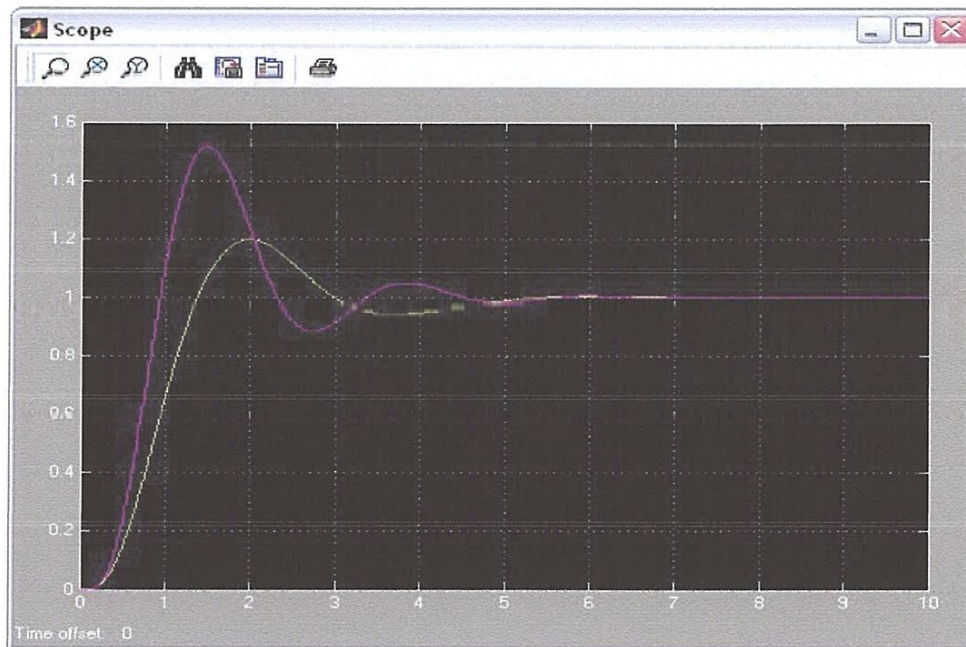


Figure IV.23: Réponses du système avec un correcteur PID et floue

Presque le même résultat de la figure (IV.9) La courbe en violet représente la réponse en échelon du système commandé par le PID, et la courbe en jaune représente la réponse en échelon du système commandé par le contrôleur flou. Les deux contrôleurs ont le même temps de stabilisation, mais le contrôleur flou a un plus petit dépassement maximal.

IV.5 Conclusion :

D'après les résultats de simulation précédent on peut déduire que la commande floue donne des résultats précieux surtout lorsque on effectue des changements de retard. Par contre la commande classique PID perd ses performances notamment en terme (stabilité, dépassement et la précision dynamique), et le régulateur floue reste le plus robuste.

Dans ce travail, nous avons présenté la commande des systèmes par la logique floue. Les lois de commande classique du type PI ou PID donnent des bons résultats dans le cas des systèmes linéaires à paramètres constants. Pour des systèmes non linéaires où ayant des paramètres non constants, ces lois de commande classique peuvent être insuffisantes car elles sont non robustes surtout lorsque les exigences sur la précision et autres caractéristiques dynamiques du système sont strictes, on doit faire appel à des lois de commande insensibles aux variations de paramètres aux perturbations et aux non linéarités.

La logique floue est une approche au raisonnement humain. On a vu les étapes et les méthodes les plus répandus pour la réalisation d'un système basé sur la logique floue. Les fonctions d'appartenance sont le cœur de la logique floue, et ces eux qui mettent la différence entre la logique classique (booléenne) et la logique floue. L'inférence est ou l'expérience humaine détermine la manière de raisonnement du système. Et l'agrégation c'est ou le système prend le rôle de l'être humain et commence à raisonner et de défuzzifier en suite pour donner des résultats.

La commande floue est une méthode nouvelle de commande des procédés industriels. Elle utilise des techniques s'appuyant sur le savoir-faire humain, exprimées en mots et en phrases ordinaires, plutôt que sur des équations.

Le but de notre travail est d'étudier en simulation l'application d'une commande de régulation de la température, puis de faire une comparaison avec la commande analogique PID du même procédé. Ce procédé est une régulation de température d'une serre.

Dans un premier temps, nous avons simulé, à l'aide du logiciel Simulink, la commande en poursuite par logique floue et par PID, les résultats de simulation ont montré que le PID perd ses performances en particulier en termes de dépassement ou précision dynamique alors le régulateur flou reste robuste vis-à-vis aux variations de ses paramètres en particulier le retard. Plus le retard augmente, plus le PID perd ses performances notamment en termes de stabilité alors que régulateur floue garde des performances correctes.

Bibliographie

- [1] Eric Magarotto « Cours de Régulation », IUT Caen, Université de Caen, 2004.
- [2] M. Rivoire, « Cours d'automatique, Tome 2 : Asservissement-régulation commande analogique », Edition Chihab, 1995.
- [3] Mohammed-Karim FELLAH, « Cours d'Asservissements Linéaires Continus »
Département d'Electrotechnique, Faculté des Sciences de l'Ingénieur, Université
Djillali Liabès – Sidi Bel-Abbès, Septembre 2007.
- [4] Zadeh, L. A. Soft computing and fuzzy logic. *IEEE Software*, 11(6), pp 48-56, 1994
- [5] Mamdani, E.H., Assilian, S. An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man Mach. Studies*, 7(1) : 1-13, 1975.
- [6] Merabti Halim, « Etude des systèmes flous à intervalle », thèse de Université de constantine, (2008).
- [7] Wei WU. Synthèse d'un contrôleur flou par Algorithme Génétique : Application au réglage dynamique des paramètres d'un système, these de doctorat de l'Université de Lille 1, 1998.
- [8] Vaidehi .V ,Monica .S , Mohamed Sheik Safeer, « A Prediction System Based on Fuzzy Logic », Proceedings of the World Congress on Engineering and Computer Science, san francisco-usa, 2008
- [9] Sabrina FAID - Sara BOURAHLI "commande par la logique floue de la machine synchrone a aimants permanents".
- [10] Franck Dernoncourt, « introduction à la logique floue », fichier électronique, www.developpez.com, Paris, Avril 2011.
- [11] Cahier technique n °191 (la logique floue)
- [12] La logique floue et ses applications, Bernadette BOUCHON-MEUNIER, Addison-Wesley, 1995.
- [13] Jantzen, Foundations of *John Wiley Fuzzy & Sons* ,2007 Control.
- [14] Lotfi BAGHLI, « Contribution à la commande de la machine asynchrone, utilisation de la logique floue, des réseaux de neurones et des algorithmes génétiques» , thèse doctoral, université de Henri Poincaré, Nancy-I, 1999.
- [15] Sugeno, M., Kang, G.T. Structure identification of fuzzy model. *Fuzzy Sets Syst.*, 28: 15- 33, 1988.

-
- [16] Takagi, T., Sugeno, M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst. Man and Cybern.*, SMC15(1) :116–132, 1985.
- [17] Madau D., D. F. (1996). Influence value defuzzification method. *Fuzzy Systems, Proceedings of the Fifth IEEE International Conference*, Vol. 3, pp. 1819-1824.
- [18] M. RABI ; Cours de la régulation industrielle, 2010, ESTF.
- [19] B. Le Ballais ; Matlab / Simulink, Application à l'automatique linéaire, 2001, ellipse.
- [20] Bousif achraf et belrhali yasser « étude en simulation d'une régulation thermique par logique floue « 2009/2010 ».
- [21] Gabriel Cormier « Utilisation de la logique floue: contrôleur en Simulink » GIND5439.