

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

UNIVERSITE MENTOURI DE CONSTANTINE
FACULTE DES SCIENCES DE L'INGENIEUR
DEPARTEMENT D'ELECTRONIQUE

N°d'ordre :.....

Série :.....

Mémoire de magistère

Présenté par :

Mr. Merabti Halim

Option : Contrôle des systèmes

Thème

Etude des systèmes flous à intervalle

Soutenu le :03/12/2008

Devant le jury:

Président :	S.Filali	Professeur	Université Mentouri Constantine
Rapporteur :	K.Belarbi	Professeur	Université Mentouri Constantine
Examineurs :	Dr B.Boutamina	Maître de conférences	Université Mentouri Constantine
	Dr F.Hachouf	Maitre de conférences	Université Mentouri Constantine

Année Universitaire 2008/2009

Remerciements

Je tiens à remercier profondément mon encadreur : le professeur ***Khaled Belarbi*** pour la confiance qu'il m'a accordée, ses encouragements, et ses précieux conseils,

Je tiens à remercier également le professeur ***Salim Filali***, docteur ***Boutamina brahim*** et docteur ***Hachouf fella***, membres de jury, de m'avoir fait l'honneur d'accepter de juger ce travail.

Merabti halim

Sommaire

Chapitre I

Introduction générale	1
-----------------------------	---

Chapitre II

Théorie de la logique floue.....	3
II.1. Introduction.....	3
II.1.1. Logique classique et logique floue.....	3
II.1.2. Historique et définition.....	3
II.2. Sous ensembles flous	4
II.2.1. Définition.....	4
II.2.2. Opérations de base sur les sous ensembles flous.....	4
II.3. Le contrôleur à logique flou	5
II.3.1. Variables linguistiques.....	5
II.3.2. Fuzzification.....	5
II.3.3. Règles floues.....	6
II.3.4. Inférence flou.....	6
II.3.4.1. Inférence floue de Mamdani.....	6
II.3.4.2. Inférence floue de Sugeno.....	7
II.3.5. Defuzzification.....	7
II.3.6. Schéma d'une commande floue.....	8

Chapitre III

La théorie des intervalles flous et le contrôleur inverse.....	9
I. Introduction	9
II. Concepts pertinents et notations.....	9
III. Résoudre l'équation floue $b * x + a = c$	10
III.1. Principe de la méthode.....	10
III.2. Implémentation de la méthode et condition d'existence de la solution.....	11
IV. Application au calcul du contrôleur inverse	13
IV .1. Principe du contrôle	13
IV.2. Calcul de la commande inverse.....	14

Chapitre IV

Résultats de simulation.....	16
IV.1. Introduction	16
IV.2. Contrôle de la concentration de la DO dans on procédé à boues activé	17
IV.2.1. Présentation du procédé	17
IV.2.1.1. Description.....	17
IV.2.1.2. Pratique courante du contrôle des procédés à boues activées	18
1. Contrôle par le débit d'air	19
2. Contrôle par les boues recyclées	19
3. Contrôle par boues rejetées	19
IV.2.1.3. Modélisation des procédés à boues activées	19
IV.2.2. Commande par logique floue du procédé	21
IV.2.2.1. Le contrôleur flou	21
IV.2.2.2. Les résultats obtenus par le contrôleur flou.....	23
IV.2.3. Contrôle inverse par intervalles flous de la concentration de la DO dans un procédé à boues activé	25
IV.2.3.1. Le modèle à intervalle flou du procédé	25
IV.3.2.2. Détermination de la commande inverse	27
IV.2.3.2. Résultats de simulation.....	27
IV.3. Contrôle de la concentration du produit dans un réacteur chimique : CSTR.....	30
IV.3.1. Présentation et modèle du procédé	30
IV.3.2. Commande par logique floue du réacteur	32
IV.3.2.1. Le contrôleur flou.....	32
IV.3.2.2. Résultats obtenus par le contrôleur flou.....	33
IV.3.3. Contrôle par intervalle flou de la concentration du produit dans un réacteur chimique (CSTR)	34
IV.3.3.1. Le modèle à intervalles flous.....	34
IV.3.3.2. Calcul et application de la commande	35

Chapitre V

Conclusion générale.....	39
--------------------------	----

Bibliographie	40
----------------------------	----

ANNEXE

Sur les algorithmes génétiques.....	41
-------------------------------------	----

Chapitre I :

Introduction générale

En automatique, un contrôleur calculé pour un modèle donné du procédé, peut lorsqu'il est implanté produire des performances assez éloigné de celles prévues et parfois même provoquer l'instabilité du système en boucle fermé. Ceci est du aux incertitudes sur le modèle et/ou aux variations de certains paramètres. Aussi la robustesse des contrôleurs est-elle devenue un important critère de la conception des systèmes commandés en boucle fermée. Celle-ci est définie alors comme la capacité de ces systèmes à garder performances et/ou stabilité dans un domaine prédéfini en débit des incertitudes et variations. Un nouveau domaine de recherche en automatique, la commande robuste, est alors apparu au milieu des années 1980 et plusieurs méthodes de conception ont alors été proposées. Le point commun de ces méthodes est la prise en compte des incertitudes sur les modèles, représentées par des intervalles de variations, dans la conception des contrôleurs [1].

Par ailleurs, les contrôleurs à logique floue, devenus classiques, ont en pratique des propriétés de robustesse assez remarquables [2] qui ne sont cependant pas étayées par des études théoriques. Il est alors intéressant d'utiliser le paradigme de base de la commande robuste c'est à dire l'intégration des incertitudes dans la phase de synthèse pour la conception de contrôleurs à logique floue robustes. Un premier pas a été effectué dans ce sens par l'utilisation de la notion d'intervalle floue [3,4,5]. Dans ce cas le modèle du procédé est représenté avec des paramètres définis sur des intervalles flous et quelques algorithmes de modélisation de systèmes dynamiques et de conception de contrôleurs relativement simples ont alors été proposés [6,7,8,9]. Le signal de commande lui-même peut être obtenu comme un intervalle flou, c'est le cas en particulier du contrôleur inverse proposé dans [8]. Dans la

phase finale de conception il s'agira alors de choisir selon un critère donné, un signal de commande contenu dans cet intervalle sachant que tous ceux-ci satisfont les performances désirées.

Dans ce travail, nous considérons l'application de la théorie des intervalles flous dans le domaine de l'identification et la commande des systèmes non linéaires. La phase d'identification consistera à construire un modèle représenté par des intervalles flous en utilisant une base de données. La construction du modèle sera réalisée à l'aide des algorithmes génétiques. Le modèle est alors utilisé pour construire des contrôleurs inverses proposés dans [8]. Nous nous proposons de comparer les performances de ces contrôleurs avec les contrôleurs à logique floue. Deux exemples sont traités, la commande d'un système de purification des eaux usées par les boues activées et la commande d'un réacteur chimique continu.

Pour cela ce mémoire est organisé comme suit : dans le second chapitre nous rappelons brièvement la théorie de la logique floue, dans le troisième chapitre nous présentons quelques éléments de la théorie des intervalles flous et l'algorithme pour la conception des contrôleurs inverses de Boukezzoula et al, enfin les applications sont présentées dans le chapitre quatre.

Chapitre II :

Théorie de la logique floue

II.1. Introduction

De nos jours, la logique floue (en anglais «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,...) que dans le domaine industriel (réglage et commande de processus complexes liés à l'énergie, aux transports, à la transformation de la matière, à la robotique, aux machines-outils)[2]. Dans ce chapitre, nous présentons les éléments de base de la théorie de la logique en insistant sur les contrôleurs à logique floue.

II.1.1. Logique classique et logique floue

Dans la logique classique, les variables gérées sont Booléennes. C'est à dire qu'elles ne prennent que deux valeurs 0 ou 1. La logique floue a pour but de raisonner à partir de connaissances imparfaites qui opposent résistance à la logique classique. Pour cela la logique floue se propose de remplacer les variables booléennes par des variables flous.

II.1.2. Historique et définition

Depuis longtemps l'homme recherche à maîtriser les incertitudes et les imperfections inhérentes à sa nature. La première réelle manifestation de la volonté de formaliser la prise en compte des connaissances incertaines fut le développement de la théorie des probabilités à partir du XVII siècle. Mais les probabilités ne peuvent maîtriser les incertitudes psychologiques et linguistiques.

Puis la logique floue est apparue en 1965 à Berkeley dans le laboratoire de Lotfi Zadeh avec la théorie des sous-ensembles flous puis en 1978 avec la théorie des possibilités. Ces deux théories constituent aujourd'hui ce que l'on appelle Logique Floue.

La logique floue permet la formalisation des imprécisions dues à une connaissance globale d'un système très complexe et l'expression du comportement d'un système par des

mots. Elle permet donc la standardisation de la description d'un système et du traitement de données aussi bien numériques qu'exprimées symboliquement par des qualifications linguistiques.

II.2. Sous ensembles flous

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.

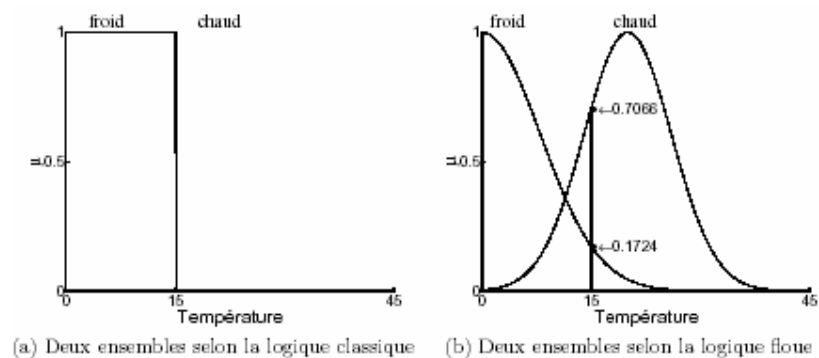


Figure II.1-Classification des températures d'une pièce en deux ensembles

II.2.1. Définition

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 $[0, 1]$:

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

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

$$A = \{ (x, \mu_A(x)) / x \in X \} \quad (\text{II.2})$$

II.2.2. Opérations de base 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'inclusion, l'intersection, l'union et le complément grâce à des opérations sur les fonctions d'appartenance.

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 :

$$\forall x \in X : \mu_A(x) \leq \mu_B(x) \quad (\text{II.3})$$

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 :

$$\forall x \in X : \mu_{A \cap B} = \min(\mu_A(x), \mu_B(x)) \quad (\text{II.4})$$

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 :

$$\forall x \in X : \mu_{A \cup B} = \max(\mu_A(x), \mu_B(x)) \quad (\text{II.5})$$

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 :

$$\forall x \in X : \mu_{A^c}(x) = 1 - \mu_A(x) \quad (\text{II.6})$$

II.3. Le contrôleur à logique flou

II.3.1. Variables linguistiques

Une variable linguistique est caractérisée par un quintuple $(V, T(V), X, G, M)$, dans lequel :

- V est le nom de la variable définie sur l'univers du discours X .
- $T(V) = A_1, A_2, \dots, A_n$ est un ensemble des termes linguistiques qui sont des nombres flous, définissant des restrictions sur les valeurs que prend V dans X .
- G est un ensemble de règles syntaxiques qui permettent de former d'autres termes linguistiques à partir de $T(V)$. On les appelle modificateurs linguistiques. Par exemple, pour définir la fonction d'appartenance du terme linguistique «pas A» on utilise l'expression

$$\mu_{pas(A)} = 1 - \mu_A \quad (\text{II.7})$$

- M est l'ensemble des règles sémantiques qui permettent de définir les termes linguistiques.

II.3.2. 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 :

$$y = 2 * c(x - (a + b) / 2) / (b - a) \quad (\text{II.8})$$

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.

II.3.3. 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 ».

II.3.4. Inférence flou

L'inférence floue est 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.

II.3.4.1. 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 processus d'inférence peut être décrit par le schéma suivant :

Règle 1 : Si (x_1 est A_{11}) et \dots et (x_m est A_{1m}) ; alors (Y est B_1)

Règle 2 : Si (x_1 est A_{21}) et \dots et (x_m est A_{2m}) ; alors (Y est B_2)

\dots

Règle n : Si (x_1 est A_{n1}) et \dots et (x_m est A_{nm}) ; alors (Y est B_n)

Dans lequel x_1, \dots, x_m sont des éléments des univers du discours X_1, \dots, X_m et $A_{ji}, (j = 1, \dots, m)$, sont des quantités floues sur l'univers du discours X_i , et $B_j, (j = 1, \dots, m)$, sont également des quantités floues sur l'univers du discours Y .

Afin de définir une seule prémisse pour une règle i, les propositions « x_j 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 :

$$\mu_{R_{A_i}}(x_1, \dots, x_m) = \mu_{A_{i1}}(x_1) \wedge \dots \wedge \mu_{A_{im}}(x_m) \quad (\text{II.9})$$

II.3.4.2. Inférence floue de Sugeno

Sugeno a proposé une méthode d'inférence floue qui garantit la continuité de la sortie. 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 univers du discours X_1, \dots, X_m et A_{i1}, \dots, A_{im} sont des termes linguistiques sur ces mêmes univers du discours. y est une fonction de x_1, \dots, x_m .

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.

II.3.5. Defuzzification

La défuzzification est le traitement qui permet de définir une correspondance entre le résultat de l'inférence et la grandeur continue fournie en sortie.

Il y'a plusieurs méthodes de défuzzification, la plus utilisé est la règle de centre de gravité :

$$y_{cg} = \left(\sum_{i=1}^m \mu_{R_i} * y_i \right) / \sum_{i=1}^m \mu_{R_i} \quad (\text{II.10})$$

II.3.6. Schéma d'une commande floue

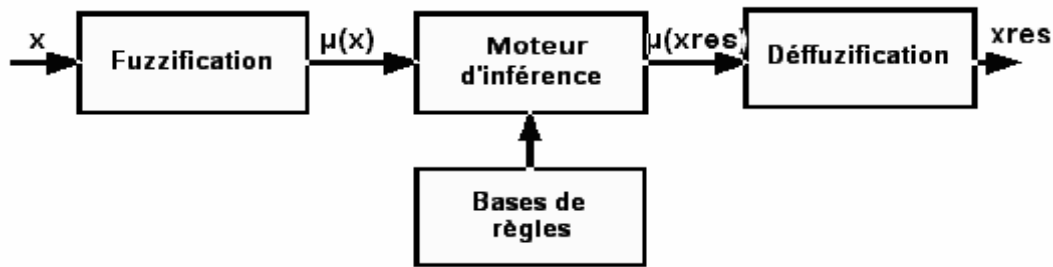


Figure II.2- Schéma d'une commande floue

La mise en oeuvre d'une commande floue fait apparaître trois grands modules.

Le premier module traite les entrées du système (valeurs réglant). On définit tout d'abord un univers de discours, un partitionnement de cet univers en classes pour chaque entrée, et des fonctions d'appartenance pour chacune de ces entrées (par exemple pression grande, petite, faible et changement d'écart mesure consigne de débit de matériau sortant d'une trémie très élevé, élevé, moyen, négatif, très négatif). La première étape, appelée fuzzification, consiste à attribuer à la valeur réelle de chaque entrée, au temps t , sa fonction d'appartenance à chacune des classes préalablement définies, donc à transformer l'entrée réelle en un sous ensemble floue. Le deuxième module consiste en l'application de règles de type «si l'écart de température est grand, diminué le débit du fuel». Ces règles vont, comme dans l'exemple introductif, permettre de passer d'un degré d'appartenance d'une grandeur réglante au degré d'appartenance d'une commande. Ce module est constitué d'une base de règles et d'un moteur d'inférence qui permet le calcul.

Le troisième et le dernier module décrit l'étape de déffuzification qui est la transformation inverse de la première. Il permet de passer d'un degré d'appartenance d'une commande à la détermination de la valeur à donner à cette commande.

Chapitre III :

La théorie des intervalles flous et le contrôleur inverse

I. Introduction

La théorie des intervalles flous introduite en particulier dans [3] a trouvé des applications intéressantes pour modéliser et commander les systèmes présentant des incertitudes dans les paramètres. Contrairement à un intervalle conventionnel qui inclut seulement toutes les valeurs possibles, un intervalle flou contient dans sa définition un degré d'incertitude pour chacune des valeurs possibles.

Dans ce chapitre basé sur [8], nous présentons la méthode de synthèse du contrôleur inverse pour les systèmes à intervalles flous. Pour cela nous introduisons dans la première section les notions de base des intervalles flous, dans la deuxième section la méthode de résolution d'une équation floue telle que décrite dans [8] enfin dans la troisième section la méthode de calcul de la commande inverse. Le but est de proposer une méthodologie de la synthèse du contrôle, basé sur l'inversion du modèle, pour les systèmes à intervalle flous.

II. Concepts pertinents et notations

Définition 1 : Intervalle conventionnel

On définit un nombre de l'intervalle a comme l'ensemble des nombres réelles x tel que :

$$a = [a^-, a^+] = \{ x / a^- \leq x \leq a^+, x \in R \} \quad (\text{III.1})$$

Pour deux intervalles $a_1 = [a_1^-, a_1^+]$ et $a_2 = [a_2^-, a_2^+]$, les opérations arithmétique sur l'intervalle (addition, soustraction, multiplication et division) peut être défini comme suit:

$$a_1 + a_2 = [a_1^- + a_2^-, a_1^+ + a_2^+] \quad (\text{III.2})$$

$$a_1 - a_2 = [a_1^- - a_2^+, a_1^+ - a_2^-] \quad (\text{III.3})$$

$$a_1 * a_2 = [\text{Min}(E_1), \text{Max}(E_1)] \quad \text{Avec } E_1 = \{ a_1^- * a_2^-, a_1^- * a_2^+, a_1^+ * a_2^-, a_1^+ * a_2^+ \} \quad (\text{III.4})$$

$$\frac{a_1}{a_2} = [a_1^-, a_1^+] * \left[\frac{1}{a_2^+}, \frac{1}{a_2^-} \right], \text{ si } 0 \notin a_2. \quad (\text{III.5})$$

On définit le point *milieu* Mid (midpoint), le *rayon* Rad (radius) et la fonction *d'ampleur relative* (zone relative) Rex d'un intervalle a comme suit:

$$Mid[a] = \frac{a^- + a^+}{2} \quad (III.6)$$

$$Rad[a] = \frac{a^+ - a^-}{2} \quad (III.7)$$

$$Et : Rex[a] = \frac{Rad[a]}{Mid[a]} \quad (III.8)$$

Définition 2 : Intervalle flou

Un intervalle flou a est constitué d'un intervalle conventionnel auquel est associé une fonction d'appartenance est dénoté μ_a , défini comme suit :

$$\mu_a : \begin{cases} R \rightarrow [0,1] \\ x \rightarrow \mu_a(x) \end{cases} \quad (III.10)$$

Chaque valeur de l'intervalle a un degré d'appartenance, plus ce degré est élevé plus la valeur est plausible.

Dans le cas où la fonction d'appartenance est une fonction triangulaire ou trapézoïdale, l'intervalle flou peut être exprimé par les relations suivantes :

$$a = [a^-, a^+] : \text{Noyau } (a) = [(a^{\text{mod}})^-, (a^{\text{mod}})^+] \text{ pour une fonction d'appartenance trapézoïdale,}$$

$$a = [a^-, a^+] : \text{Noyau } (a) = [a^{\text{mod}}] \text{ pour une fonction d'appartenance triangulaire.}$$

Le noyau est composé par les valeurs les plus plausibles des paramètres.

Définition 3 : α -coup

Une α -coup de l'intervalle flou peut être définie comme suit :

$$a^\alpha = \{x / \mu_a(x) \geq \alpha; \alpha \in [0,1]\} \quad (III.11)$$

Et peut être aussi donné sous l'expression d'un intervalle conventionnel :

$$a^\alpha = [(a)^\alpha]^-, [(a)^\alpha]^+ \quad (III.12)$$

III. Résoudre l'équation floue $b * x + a = c$:

Avant de détailler la stratégie du contrôle, cette section présente le formalisme associé avec la résolution de l'équation floue qui est fondamental dans la méthodologie de conception du contrôle.

III.1. Principe de la méthode

Soit a , b et c des intervalles flous avec $0 \notin b$.

L'équation floue $b * x + a = c$ peut être exprimée par α -coupes de a , b , c et x qui mènent à l'équation de l'intervalle suivante:

$$b^\alpha * x^\alpha + a^\alpha = c^\alpha \quad (\text{III.13})$$

Avec : $\alpha \in [0,1]$ Définit α -coupes pour α dans $[0,1]$

$$a^\alpha = [(a^\alpha)^-, (a^\alpha)^+]; b^\alpha = [(b^\alpha)^-, (b^\alpha)^+]; c^\alpha = [(c^\alpha)^-, (c^\alpha)^+] \quad (\text{III.14})$$

Comme la soustraction et la division ne sont pas les opérations symétriques à l'addition et la multiplication, la solution exacte de (III.13) n'est pas

$$x^\alpha = \frac{(c^\alpha - a^\alpha)}{b^\alpha} \quad (\text{III.15})$$

Dans la section suivante, on présente une méthode de solution pour cette équation [9].

III.2. Implémentation de la méthode et condition d'existence de la solution

La méthode proposée est implémentée selon les deux étapes suivantes :

➤ Résoudre l'équation (III.13) sachant que $d^\alpha = b^\alpha * x^\alpha$. (III.16)

L'objectif ici est de calculer la solution exacte d^α de l'équation $d^\alpha = b^\alpha * x^\alpha$

La solution exacte de (III.16) est donné par:

$$d^\alpha = [(d^\alpha)^-, (d^\alpha)^+] = [(c^\alpha)^- - (a^\alpha)^-, (c^\alpha)^+ - (a^\alpha)^+] \quad (\text{III.17})$$

Une condition suffisante pour l'existence de la solution est donner par :

$$(d^\alpha)^- \leq (d^\alpha)^+ \Leftrightarrow Rad(a^\alpha) \leq Rad(c^\alpha). \quad (\text{III.18})$$

➤ Résoudre l'équation initiale, dont le variable est d , ensuite tirer x .

L'objectif ici est de résoudre l'équation initiale (dont le variable est d) prenant en considération x .

$$\text{On a } d^\alpha = b^\alpha * x^\alpha = b^\alpha * [(x^\alpha)^-, (x^\alpha)^+]. \quad (\text{III.19})$$

On distingue deux cas :

1. si $b^\alpha > 0$

Les expressions de $(d^\alpha)^-$ et $(d^\alpha)^+$ sont données par :

$$(d^\alpha)^+ = (b^\alpha * x^\alpha)^+ = Mid(b^\alpha) * (x^\alpha)^+ + Rad(b^\alpha) * |(x^\alpha)^+| \quad (\text{III.20})$$

$$(d^\alpha)^- = (b^\alpha * x^\alpha)^- = \text{Mid}(b^\alpha) * (x^\alpha)^- - \text{Rad}(b^\alpha) * |(x^\alpha)^-| \quad (\text{III.21})$$

Dans ce cas, comme

$$\text{sign}[(x^\alpha)^-] = \text{sign}[(d^\alpha)^-] \text{ et } \text{sign}[(x^\alpha)^+] = \text{sign}[(d^\alpha)^+] \quad (\text{III.22})$$

Les équations (III.20) et (III.21) peuvent être réécrites sous la forme :

$$\begin{cases} (d^\alpha)^- = [\text{Mid}(b^\alpha) - \text{Rad}(b^\alpha) * \text{sign}[(d^\alpha)^-]] * (x^\alpha)^- \\ (d^\alpha)^+ = [\text{Mid}(b^\alpha) + \text{Rad}(b^\alpha) * \text{sign}[(d^\alpha)^+]] * (x^\alpha)^+ \end{cases} \quad (\text{III.23})$$

Finalement, on trouve :

$$\begin{cases} (x^\alpha)^- = \frac{(d^\alpha)^-}{\text{Mid}(b^\alpha) - \text{Rad}(b^\alpha) * \text{sign}[(d^\alpha)^-]} \\ (x^\alpha)^+ = \frac{(d^\alpha)^+}{\text{Mid}(b^\alpha) + \text{Rad}(b^\alpha) * \text{sign}[(d^\alpha)^+]} \end{cases} \quad (\text{III.24})$$

2. si $b^\alpha < 0$

Les expressions de $(d^\alpha)^-$ et $(d^\alpha)^+$ sont données par :

$$\begin{cases} (d^\alpha)^- = (b^\alpha * x^\alpha)^- = \text{Mid}(b^\alpha) * (x^\alpha)^+ - \text{Rad}(b^\alpha) * |(x^\alpha)^+| \\ (d^\alpha)^+ = (b^\alpha * x^\alpha)^+ = \text{Mid}(b^\alpha) * (x^\alpha)^- + \text{Rad}(b^\alpha) * |(x^\alpha)^-| \end{cases} \quad (\text{III.25})$$

On trouve les résultats suivants :

$$\begin{cases} (x^\alpha)^- = \frac{(d^\alpha)^+}{\text{Mid}(b^\alpha) - \text{Rad}(b^\alpha) * \text{sign}[(d^\alpha)^+]} \\ (x^\alpha)^+ = \frac{(d^\alpha)^-}{\text{Mid}(b^\alpha) + \text{Rad}(b^\alpha) * \text{sign}[(d^\alpha)^-]} \end{cases} \quad (\text{III.26})$$

Proposition

La condition d'existence de solution peut être écrit sur la forme suivante :

$$\text{Rad}[c^\alpha] \geq \text{Rad}[a^\alpha] + |\text{Re}x(b^\alpha)| * |\text{Mid}[c^\alpha] - \text{Mid}[a^\alpha]| \quad (\text{III.27})$$

La démonstration se trouve dans [9].

IV. Application au calcul du contrôleur inverse :**IV .1. Principe du contrôle :**

On considère un système linéaire, stable, invariant dans le temps, contrôlable, simple entrée, simple sortie, dont sa représentation mathématique est donné par :

$$y(k+1) = \sum_{i=1}^n a_i * y(k-i+1) + \sum_{j=0}^{m-1} b_j * u(k-j) ; \quad m < n \quad (\text{III.28})$$

Où $y(k)$ est la sortie du système et $u(k)$ correspond à l'entrée du système,

$a_i, i=1, \dots, n ; b_j ; j=0, \dots, m-1$ représente les paramètres du système. On suppose que les paramètres du système sont des intervalles flous.

On peut considérer, de la propriété de la contrôlabilité du système, que $0 \notin b_0$, c'est-à-dire que le signe de b_0 est connu. Dans ce cas, l'objectif du contrôle est de forcer la sortie du système de suivre une trajectoire floue désirée y_d .

On considère une α -coupe sur les paramètres du système (intervalle flou) a_i, b_j ; on peut écrire l'intervalle flou du système (III.28) comme suit :

$$y^\alpha(k+1) = \sum_{i=1}^n a_i^\alpha * y^\alpha(k-i+1) + \sum_{j=1}^{m-1} b_j^\alpha * u^\alpha(k-j) + b_0^\alpha * u^\alpha(k) \quad (\text{III.29})$$

Dont les paramètres sont donnés par : $a_i^\alpha = [(a_i^\alpha)^-, (a_i^\alpha)^+]$; $b_j^\alpha = [(b_j^\alpha)^-, (b_j^\alpha)^+]$.

Pour simplifier l'écriture, la relation (III.29) est reformuler dans (III.30) :

$$y^\alpha(k+1) = \psi^\alpha[Z(k)] + b_0^\alpha * u^\alpha(k) \quad (\text{III.30})$$

Dont

$$\psi^\alpha[Z(k)] = \sum_{i=1}^n a_i^\alpha * y^\alpha(k-i+1) + \sum_{j=1}^{m-1} b_j^\alpha * u^\alpha(k-j) = [\psi^\alpha[Z(k)]^-, \psi^\alpha[Z(k)]^+] \quad (\text{III.31})$$

$$\text{et } Z(k) = [y^\alpha(k), \dots, y^\alpha(k-n+1), u^\alpha(k-1), \dots, u^\alpha(k-m+1)]. \quad (\text{III.32})$$

Nous considérons ici des systèmes du contrôle en boucle fermé, alors si on suppose que les paramètres du système sont des intervalles flous, alors il faut que la trajectoire de la référence devrait être aussi floue. Dans ce cas, le dernier devrait être considéré comme l'objectif flou à atteindre. Par exemple, si la trajectoire désirée est choisie comme un intervalle flou trapézoïdal, le niveau d'appartenance est considéré comme un degré de préférence.

En effet, tout α -coupe de la référence floue est l'ensemble de toutes les valeurs qui satisfont l'objectif du contrôle. Le noyau (kernel) est composé par les valeurs les plus acceptables.

Dans ce cas, l'objectif du contrôle peut être envisagé comme la détermination d'un contrôleur capable de maintenir la réponse du système(y) dans une enveloppe de tolérance autour de la trajectoire exacte y_d^{mod} , c à d

$$y \in [y_d^{\text{mod}} - \Delta^\alpha, y_d^{\text{mod}} + \Delta^\alpha] \quad (\text{III.33})$$

Où Δ^α est la borne autour de la trajectoire exacte.

Dans ce cas, pour un degré de préférence α donné sur la trajectoire floue désirée, l'objectif du contrôle est de concevoir un contrôleur, capable de maintenir la sortie du système dans une enveloppe autour de la trajectoire désirée,

$$y^\alpha(k+1) = [(y^\alpha(k+1))^-, (y^\alpha(k+1))^+] = [y_d^{\text{mod}}(k+1) - \Delta^\alpha, y_d^{\text{mod}}(k+1) + \Delta^\alpha] \quad (\text{III.34})$$

Dans cette structure, le problème du contrôle peut être adressé en résolvant l'équation :

$$[y_d^{\text{mod}}(k+1) - \Delta^\alpha, y_d^{\text{mod}}(k+1) + \Delta^\alpha] = \psi^\alpha[Z(k)] + b_0 * u^\alpha(k) \quad (\text{III.35})$$

IV.2. Calcul de la commande inverse

D'après le paragraphe précédent, l'expression de l'entrée du contrôle est donnée par :

$$\begin{cases} (u^\alpha(k))^- = \frac{N_1}{D_1} = \frac{\text{Mid}[v^\alpha(k)] - \text{Rad}[v^\alpha(k)] * \text{sign}[\text{Mid}(b_0^\alpha)]}{\text{Mid}(b_0^\alpha) - \text{Rad}(b_0^\alpha) * \text{sign}[N_1]} \\ (u^\alpha(k))^+ = \frac{N_2}{D_2} = \frac{\text{Mid}[v^\alpha(k)] + \text{Rad}[v^\alpha(k)] * \text{sign}[\text{Mid}(b_0^\alpha)]}{\text{Mid}(b_0^\alpha) + \text{Rad}(b_0^\alpha) * \text{sign}[N_2]} \end{cases} \quad (\text{III.36})$$

Dont :

$$\begin{cases} (v^\alpha(k))^- = y_d^{\text{mod}}(k+1) - \Delta^\alpha - (\psi^\alpha[Z(k)])^- \\ (v^\alpha(k))^+ = y_d^{\text{mod}}(k+1) + \Delta^\alpha - (\psi^\alpha[Z(k)])^+ \end{cases} \quad (\text{III.37})$$

Et

$$\begin{cases} \text{Mid}[v^\alpha(k)] = y_d^{\text{mod}}(k+1) - \text{Mid}(\psi^\alpha[Z(k)]) \\ \text{Rad}[v^\alpha(k)] = \Delta^\alpha - \text{Rad}(\psi^\alpha[Z(k)]) \end{cases} \quad (\text{III.38})$$

Ici la condition d'existence du contrôleur inversé est donnée par l'inégalité suivante :

$$\text{Rad}[y_d^{\text{mod}}(k+1) - \Delta^\alpha, y_d^{\text{mod}}(k+1) + \Delta^\alpha] \geq \text{Rad}[\psi^\alpha[Z(k)]] + |\text{Re } x(b_0^\alpha)| * |\text{Mid}[v^\alpha(k)]| \quad (\text{III.39})$$

La démonstration est dans [9].

Chapitre IV :

Résultats de simulation

IV.1. Introduction :

Dans ce chapitre, nous considérons l'application du contrôleur inverse décrit dans le chapitre précédent à deux systèmes en l'occurrence un procédé à boues activées et un réacteur chimique CSTR. Nous développons en même temps une commande par logique floue pour comparaison.

La commande inverse basée sur les intervalles flous, nécessite des modèles linéaires avec des paramètres définis sur des intervalles flous. Pour obtenir le contrôleur inverse il faut développer les étapes suivantes :

- Déterminer le modèle mathématique linéaire de ce système (modèle entrée- sortie)
- Déterminer les intervalles des paramètres de ce modèle en utilisant les algorithmes génétiques
- Déterminer la fonction d'appartenance des paramètres
- Déterminer le degré de l'incertitude sur la référence (l'incertitude sur la trajectoire désirée y_d)
- Déterminer le degré de l'incertitude sur les paramètres du système,
- Calculer les valeurs de la commande (déterminer l'intervalle de la commande par les formules du chapitre II),
- Appliquer une seule commande au système selon une loi de commande (ou régulation),

Dans ce travail, les fonctions d'appartenance des intervalles sont triangulaire avec $\alpha = 0$ (cf. chapitre III).

IV.2. Contrôle de la concentration de la DO dans un procédé à boues activé :

IV.2.1. Présentation du procédé :

IV.2.1.1. Description

Par souci d'économie énergétique et d'amélioration du taux et de la qualité des eaux épurées, l'automatisation des stations de traitement des eaux usées a connu son apogée lors de l'élaboration d'instrument de mesures très sophistiqués. Le contrôle se fait alors en

manipulant des mesures telles que la demande biochimique en oxygène (BOD), la demande chimique en oxygène (COD), le carbone organique total (TOC).

Les procédés d'épurations des eaux sont des systèmes à non linéarité très marquée à cause de variations continues et brusques aussi bien en concentration qu'en composition des influents qui ont des constantes de temps différentes, ce qui complique la tâche de contrôle. Dans cette partie, on présente le contrôle de l'oxygène dissous, DO, dans le procédé de traitement biologique par boues activées aérobie. La grandeur de commande sera le débit d'air.

Les stations d'épuration des eaux usées sont diverses, celle recommandée par la pratique algérienne est schématisée par la figure IV.1:

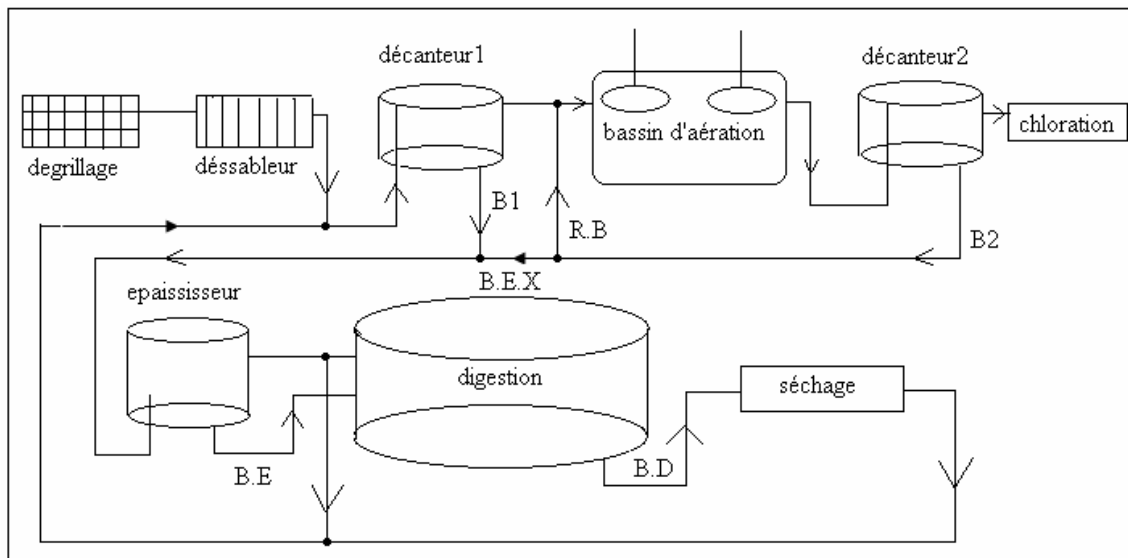


Figure IV.1-Schémas synoptique d'une station d'épuration des eaux

- Avec : B1 : boues primaires,
 B2 : boues secondaires,
 B.E : boues épaissies ;
 B.D : boues digérées ;
 R.B : retour des boues ;
 B.Ex : boues en excès.

Les différents traitements que subissent les eaux usées dans une unité industrielle de traitement sont :

Dégrillage : le but du dégrillage est d'assurer la protection des équipements électromécaniques et de réduire les risques de colmatage des conduites mises en place dans la station d'épuration.

Dessablage : qui a pour rôle de récupérer des particules organiques et les graisses qui sont sédimentées et de permettre la séparation solide liquide.

Décanteur primaire : le décanteur primaire est un moyen de réduction des contraintes de pollution qui gênent le traitement biologique, en particulier les graisses, les écumes et la DBO.

Bassin d'aération : c'est le réacteur biologique proprement dit, il reçoit les eaux résiduaires. On y injecte les boues activées et l'oxygène pour que l'opération de dégradation des éléments polluants s'effectue.

Décanteur secondaire : il permet la séparation des boues de l'influent, dont une partie est retournée au bassin d'aération alors que l'excès est évacué vers les parties du traitement des boues.

Bassin de contact : il est prévu pour la désinfection de la population bactérienne de l'effluent par l'addition de l'hypochlorite de sodium.

Les principales opérations intervenant dans le traitement par les boues activées sont :

Epaisseur : qui a principalement pour but de réduire le volume des boues en favorisant la séparation partielle eau- boue, l'échappement de gaz et l'amélioration de la consolidation.

Digesteur : c'est une opération de stabilisation qui demande des temps de séjour très important. Elle est utilisée pour le renouvellement des boues activées. La digestion aérobie est similaire au processus des boues activées, cependant la source d'énergie n'est plus le substrat mais les tissus cellulaires des micros- organismes, ce qui correspond à la phase endogène.

Lit de séchage : le séchage est une opération de séparation solide- liquide, qui utilise le principe de la percolation et le principe de l'évaporation de l'eau.

IV.2.1.2. Pratique courante du contrôle des procédés à boues activées :

La majorité des schémas de commande sont inspirés de la théorie de contrôle conventionnel, basé souvent sur des hypothèses d'aspect stationnaire. Les variables déterminant l'action de contrôle dans les procédés à boues activées sont : le débit d'air, le débit des boues recyclées, le débit des boues rejetées.

1. Contrôle par le débit d'air :

La régulation du débit d'air a une grande importance économique, cependant l'efficacité de ce paramètre est limitée par des niveaux faibles de la DO (1-2mg/l).

2. Contrôle par les boues recyclées :

Le débit des boues recyclées est proportionnel au débit d'influent. Il est utilisé pour maintenir une concentration d'oxygène dissous dans l'aérateur. Dans cette stratégie qui ne tient pas compte des variations de concentrations de l'influent, les actions de contrôle sont très lentes et peuvent affecter la qualité de l'effluent.

3. Contrôle par boues rejetées :

Cette action dépend essentiellement du temps de séjour des boues (production des boues). Elle détermine la nature et l'âge des boues. L'âge des boues (rapport entre les boues dans le système et boues produites par jour) est le critère le plus utilisé pour estimer les boues en excès. Cette technique est facile à mettre en œuvre, mais son aspect statique peut rendre l'action inefficace surtout pour des perturbations importantes. Alors une régulation du taux de croissance spécifique des boues (μ) par action sur le débit de rejet et à partir des mesures de la DO et de la masse des boues est peut être réalisée par un algorithme de type PID.

IV.2.1.3. Modélisation des procédés à boues activées :

Le fonctionnement de l'unité secondaire des stations d'épuration biologique des eaux donnée par la figure IV.2, dans laquelle les eaux résiduaires (influent) sont mises en contact avec des micro-organismes (boues activées) au niveau de l'aérateur qui peut être ouvert (aérobie) ou fermé (anaérobie). Le résultat de ce contact en présence de l'oxygène est la production des réactions de biodégradation de la pollution organique (substrat). L'oxygène nécessaire pour cette opération est fourni soit par injection (anaérobie), soit par brassage (aérobie). Le mélange passe ensuite dans le décanteur secondaire ; l'eau épurée (effluent) s'écoule par la surface, tandis que les boues plus denses sédimentent. Une partie des boues est recyclée vers l'aérateur et le reste est extrait du système. La qualité de l'effluent épuré est caractérisée par sa concentration en pollution organique (DBO) et sa concentration en boues (MLSS). Ces deux concentrations ne doivent pas dépasser en moyenne les valeurs fixées par les normes de rejet admises.

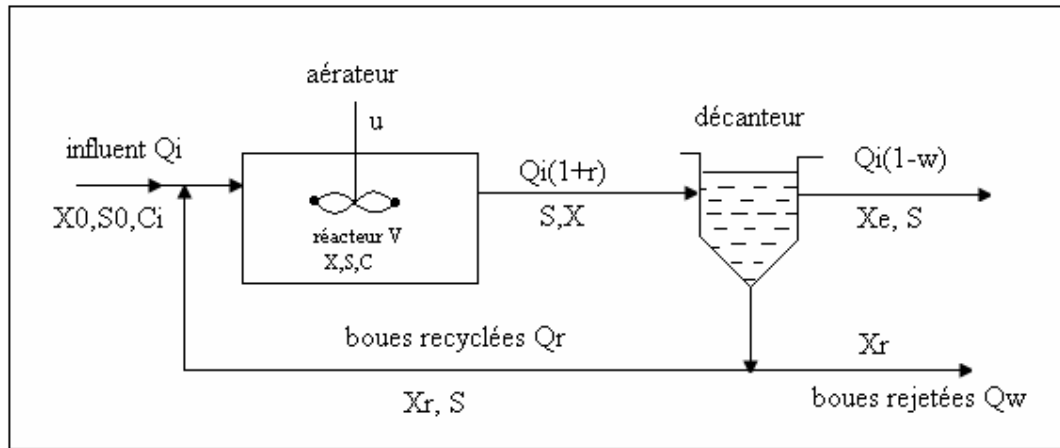


Figure IV.2-Schémas du procédé à boues activées

Dans un bio réacteur ouvert, la population bactérienne consomme, pour son développement, l'oxygène qui lui est fourni par une aération appropriée. La dynamique de la concentration en oxygène dissous DO peut être déterminée par les quatre mécanismes de bases suivantes :

- 1-la dispersion de la DO dans le bio réacteur.
- 2-Le transfert de l'oxygène de la phase gazeuse à la phase liquide.
- 3-La saisie d'oxygène pour la synthèse des cellules.
- 4-La saisie d'oxygène pour la décomposition des cellules.

Sur cette optique, la concentration de la DO est décrit comme suit :

$$\frac{dC}{dt} = \frac{Q_i}{V} * (C_i - (1+r) * C) + K_1 * U * (C_s - C) - K_2 * \mu * X - K_3 * X \quad (IV.1)$$

$$\mu = \mu_{\max} * \frac{S}{S + K_s} \quad (IV.2)$$

$$\mu_{\max} = \frac{C}{C + K_c} \quad (IV.3)$$

variables	Signification	Valeur initiale
C	Concentration de la DO	2 mg/l
Cs	Concentration de saturation de la DO	9.17 mg/l
Ci	Concentration de la DO dans l'influent	0.1 mg/l
X	Concentration des micro- organismes	150 mg/l
S	Concentration du substrat	142 mg/l
Qi	Débit d'influent	4.2 m ³

V	Volume du réacteur	1000 m^3
U	Débit d'air	0.1 m^3/min
$K1$	Coefficient de transfert d'oxygène	0.65
$K2$	Constante	0.00012
$K3$	Constante	7^{-5}
Kc	Constante de limitation	0
Ks	Constante de limitation	70
μ	Taux spécifique de croissance	0.4
μ_{\max}	Rapport de croissance maximale	0.009
dt	Le pas de RUNGE KUTTA	1min
r	Rapport entre le débit recyclé et le débit de l'influent	0.78
w	Rapport entre le débit rejeté et le débit d'influent	0.01
Ky	Coefficient de création	0.6
Kd	Constante expérimentale	0.06
$S0$	Concentration du substrat à l'entrée	270 mg/l

Tableau IV.1.paramètres du réacteur

La croissance bactérienne dans un bio réacteur alimenté en continu et infiniment mélangé est le plus souvent représentée par les équations suivantes :

$$\frac{dX}{dt} = -\frac{Q_i}{V} * (1+r) * X + \frac{Q_i}{V} * r * X_r - K_d * X + \mu * X \quad (\text{IV.5})$$

$$\frac{dS}{dt} = -\frac{Q_i}{V} * (1+r) * S + \frac{Q_i}{V} * r * S - \frac{\mu}{K_y} * X \quad (\text{IV.6})$$

L'équilibre massique au niveau du décanteur est donné par :

$$X_r = ((1+r) * X - (1-w) * X_e) / (r+w) \quad (\text{IV.7})$$

$$X_e = K_a * (1+r) * Q_i * X \quad (\text{IV.8})$$

Le modèle complet représenté par ces équations est utilisé pour la simulation.

IV.2.2. Commande par logique floue du procédé :

IV.2.2.1. Le contrôleur flou

Le contrôleur flou de Mamdani est basé sur un ensemble de règles du genre Si- Alors.

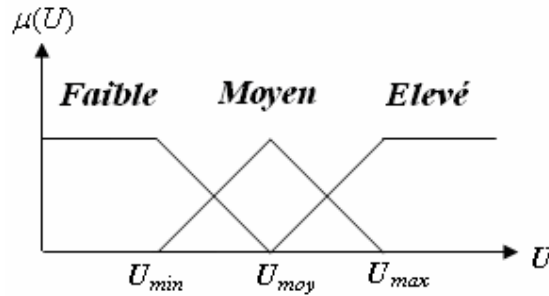
Les entrées de notre contrôleur flou sont :

e : erreur entre la sortie du système et la référence ($y(i)-ref$).

de : différence entre l'erreur actuelle et l'erreur précédente ($e(i)-e(i-1)$).

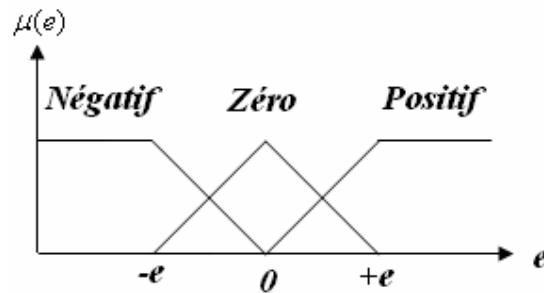
L'univers de discours et la fonction d'appartenance de chaque variable sont données par :

La commande U :



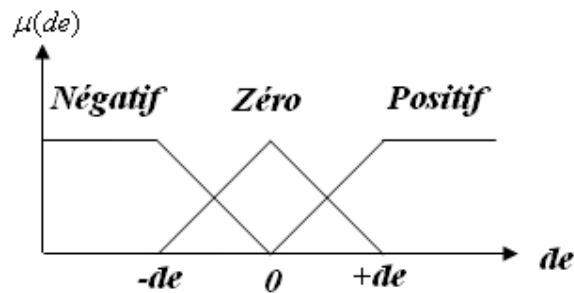
Fonction d'appartenance de la commande U

L'erreur e :



Fonction d'appartenance de l'erreur e

Variation de l'erreur :



Fonction d'appartenance de la différence d'erreur de

La base des règles est donnée dans le tableau suivant :

<i>de</i> \ <i>e</i>	<i>Négatif</i>	<i>Zéro</i>	<i>Positif</i>
<i>Négatif</i>	<i>faible</i>	<i>faible</i>	<i>Moyen</i>
<i>Zéro</i>	<i>faible</i>	<i>moyen</i>	<i>Elevé</i>
<i>Positif</i>	<i>moyen</i>	<i>élevé</i>	<i>Elevé</i>

TableauIV.2 : base de règles

IV.2.2.2. Les résultats obtenus par le contrôleur flou:

1°) Cas nominal avec référence variable

Les résultats obtenus par le contrôleur flou pour une référence variable du système sont donnés par les figures IV.3 et IV.4 pour les paramètres nominaux (aucune variation sur les paramètres internes du système). Pour la référence 6.7 l'erreur est négligeable, elle devient assez importante pour les autres cas. Notons la difficulté pour trouver un contrôleur qui donne des résultats satisfaisants pour la gamme de consigne choisie.

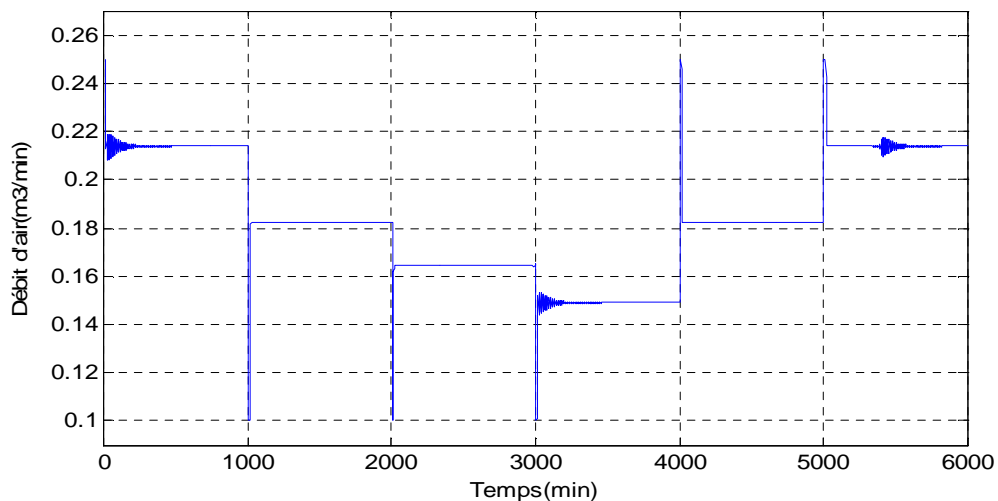


Figure IV.3-Signal de commande délivré par le contrôleur flou

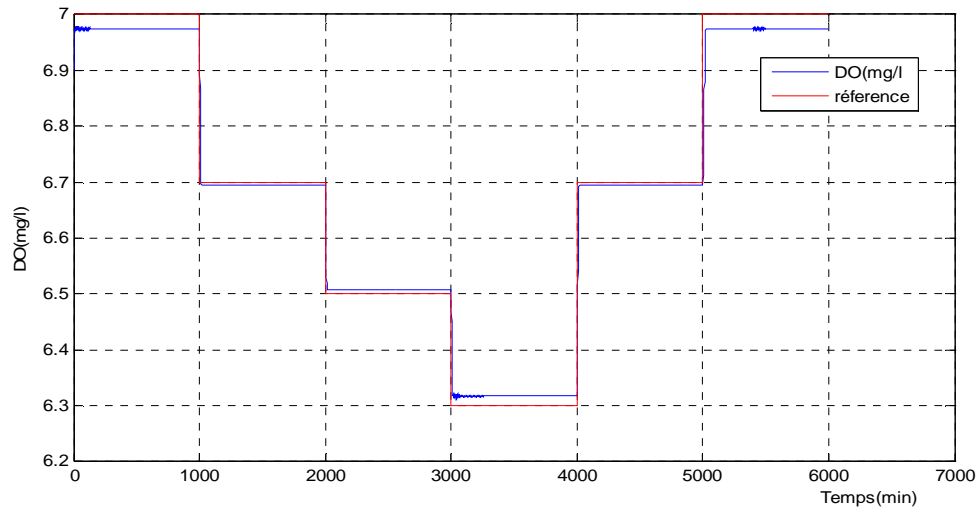


Figure IV.4-Sortie du système et référence

2°) Variation de paramètres

Pour analyser empiriquement la robustesse du contrôleur flou par rapport à des variations de paramètres, on introduit des incertitudes $\pm 10\%$ sur KI et C ($k1 = k1 \pm 0.6$; $Cs = Cs \pm 0.8$). Les résultats de simulation sont donnés par les figures IV.5 pour la commande et IV.6 pour la sortie. On remarque que la stabilité est maintenue, mais la précision se dégrade légèrement. L'intervalle de la commande du système est représenté dans la figure IV.5 :

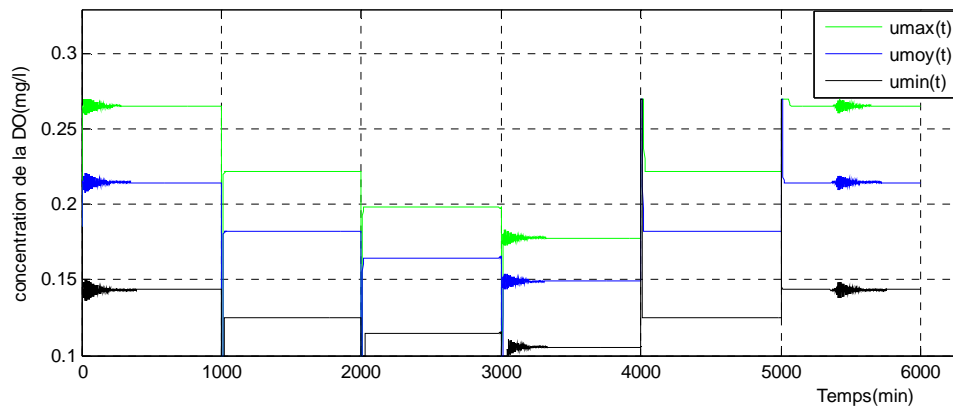


Figure IV.5- L'intervalle du signal de commande délivré par le contrôleur flou

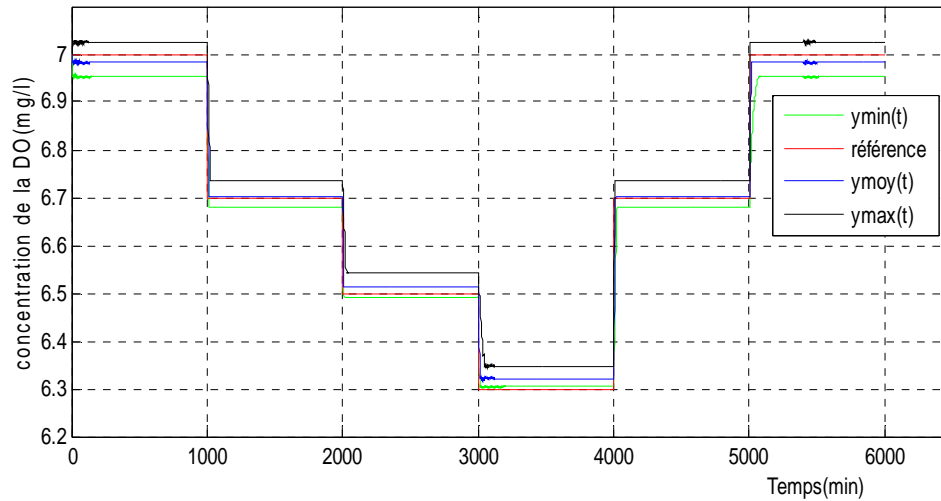


Figure IV. 6 -L'intervalle de la sortie du système par la commande floue

IV.2.3. Contrôle inverse par intervalles flous de la concentration de la DO dans un procédé à boues activé :

IV.2.3.1. Le modèle à intervalle flou du procédé :

Le problème de modélisation des systèmes est un problème d'identification qui peut être schématisé dans la figure IV.7 :

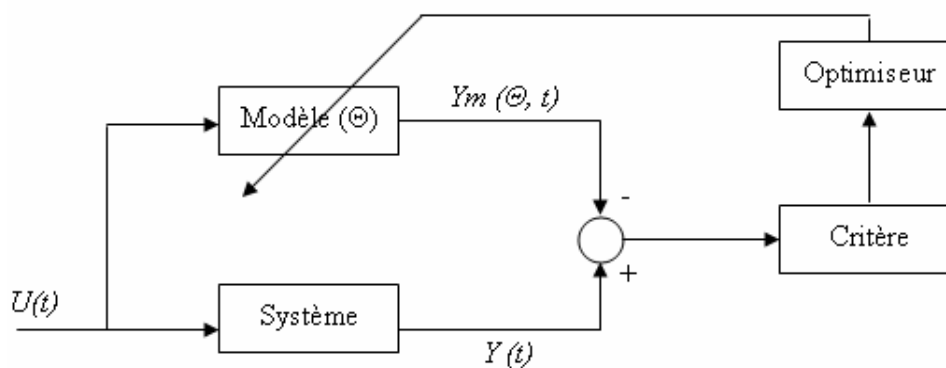


Figure IV.7-Schéma de base de l'identification

A partir de ce schéma, pour identifier un système il faut :

- Définir un critère qui permet de faire correspondre Y_m et Y
- choisir une technique d'optimisation
- choisir une forme de modèle pour ce système.

Concernant notre travail, le critère choisi est l'erreur minimale entre la sortie du système et la sortie du modèle. La technique d'optimisation est l'optimisation par un algorithme génétique simple. La forme du modèle mathématique choisi pour ce procédé est :

$$ym(i+1)=a*ym(i)+b*u(i)+c \quad (IV.9)$$

Dont :

$$a \in [a_m, a_p] = [(a)^-, (a)^+] \quad ; \quad b \in [b_m, b_p] = [(b)^-, (b)^+] \quad ; \quad c \in [c_m, c_p] = [(c)^-, (c)^+]$$

Pour déterminer les valeurs de $a_m, a_p, b_m, b_p, c_m, c_p$ on a utilisé un algorithme génétique qui peut déterminer jusqu'à 12 paramètres. Le programme donne les meilleurs paramètres qui minimisent l'erreur entre la sortie du système et la sortie du modèle pour plusieurs générations de populations.

Les valeurs obtenues, pour $\alpha=0$, sont données dans le tableau VI.3 :

paramètre	valeur
a_p	0.83844
a_m	0.82961
b_p	1.84315
b_m	1.67885
c_p	0.87042
c_m	0.69736

TableauVI.3- Paramètres du modèle du procédé1

Les fonctions d'appartenance des paramètres sont données dans la figure IV.8

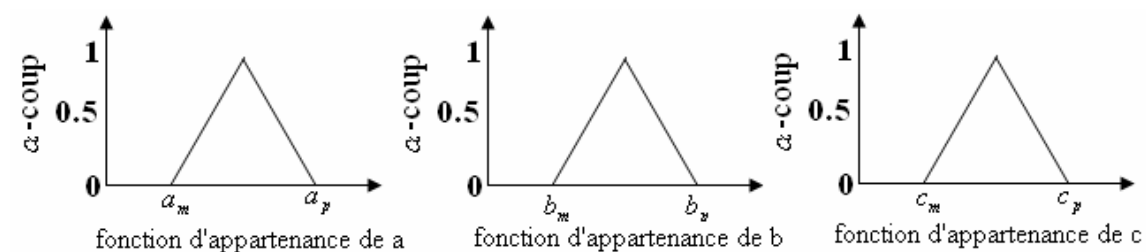


Figure IV.8- Les intervalles flous

La figure IV.9 présente la sortie du système et celle du modèle à intervalle flous.

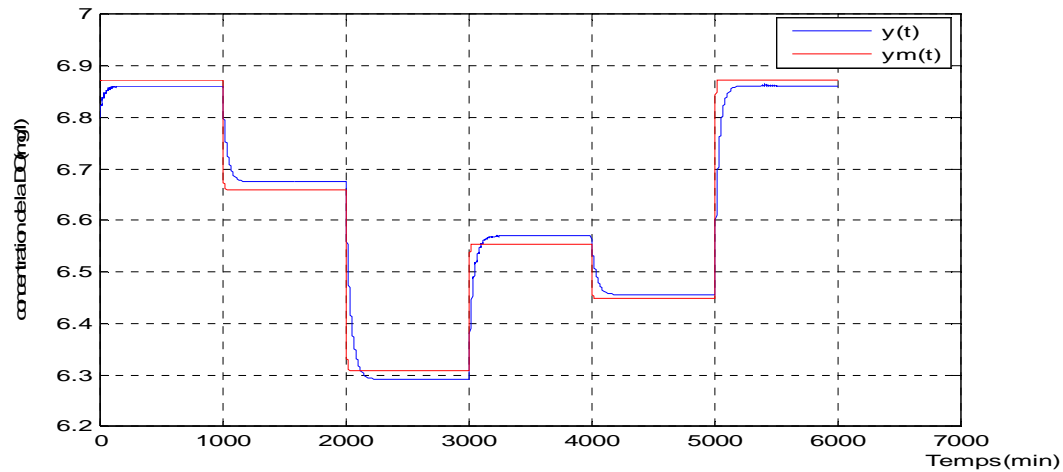


Figure IV.9-Sortie du système et du modèle

IV.2.3.2. Détermination de la commande inverse :

La commande inverse est calculée à partir des résultats donnés dans le chapitre III, on trouve :

$$um(i) = dm / (b - radb * sign(dm)) \quad (IV.10)$$

$$up(i) = dp / (b + radb * sign(dp)) \quad (IV.11)$$

avec:

$$dm = ydm(i+1) - (am * ym(i) + cm) \quad (IV.12)$$

$$dp = ydp(i+1) - (ap * yp(i) + cp) \quad (IV.13)$$

et :

$$ydp(i+1) = yd(i+1) + dlty \quad (IV.14)$$

$$ydm(i+1) = yd(i+1) - dlty \quad (IV.15)$$

y_d : est la trajectoire désiré exacte,

$dlty$: est l'incertitude sur la trajectoire désiré.

IV.2.3.3. Résultats de simulation :

La commande appliquée au système est la suivante :

$$U = ud(i) + (ey/dlty) * du \quad (IV.16)$$

avec

$$du = (up(i) - um(i)) / 2 \quad (IV.17)$$

$$ud(i) = (um(i) + up(i)) / 2 \quad (IV.18)$$

$$ey=yd(i)-c(i) \quad (IV.19)$$

$$radb=(bp-bm)/2 \quad (IV.20)$$

1°) Cas nominal

La figure IV.10 représente $yd(t)$ et $y(t)$ sans aucune incertitude sur les paramètres du système (paramètres nominaux). On remarque un bon régime transitoire avec des erreurs statiques de l'ordre de 1% ce qui est tout à fait satisfaisant pour ces systèmes. Les résultats en nominal sont clairement meilleurs que le contrôleur flou ci-dessus (cf. figure IV. 4).

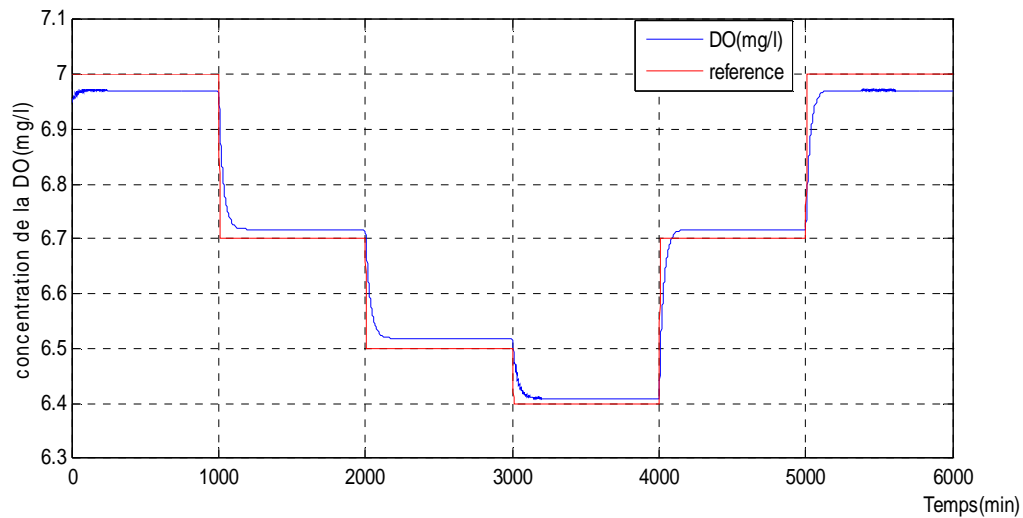


Figure IV.10-Sortie du système et référence

2°) Cas avec incertitudes

On introduit des incertitudes sur kI , Cs . telles que données sur le tableau IV.4. Les résultats de simulation sont donnés sur les figures IV.12 à IV.14 qui montrent les enveloppes de variations de la sortie et de la commande. On remarque que la commande reste bornée et acceptable tandis que la référence reste dans le domaine de variation choisi.

Domaine de yd	Intervalle de kI	Intervalle de Cs
10%	[0.59 , 0.71]	[8.4 , 10.03]
5%	[0.61 , 0.69]	[8.66 , 9.75]
1%	[0.63 , 0.685]	[8.83 , 9.5]

Tableau IV.4 – Intervalles des paramètres du système

:

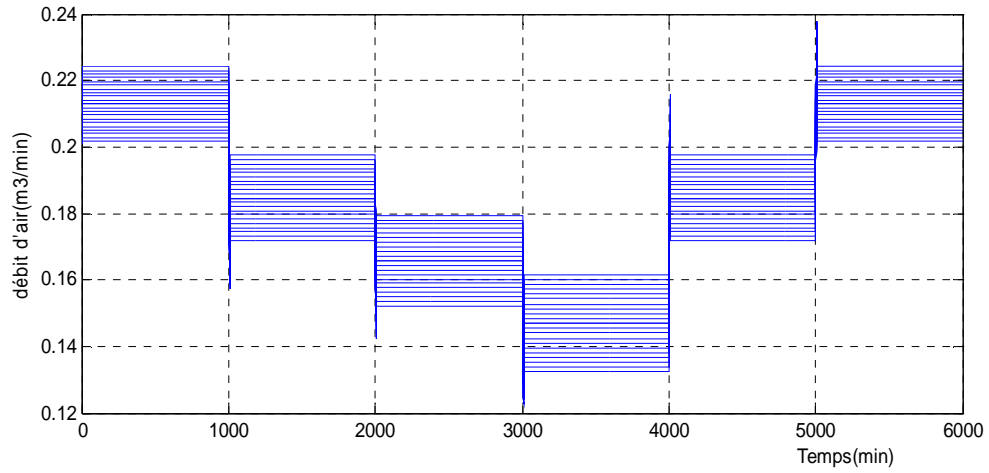


Figure IV.11- Intervalle de commande pour une incertitude de 10%

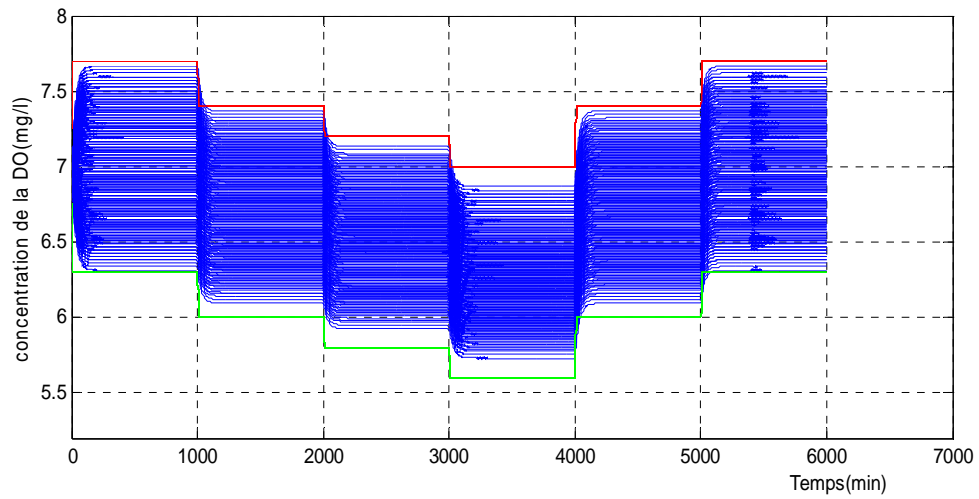


Figure IV.12-Sortie du système avec une incertitude sur les paramètres de 10%

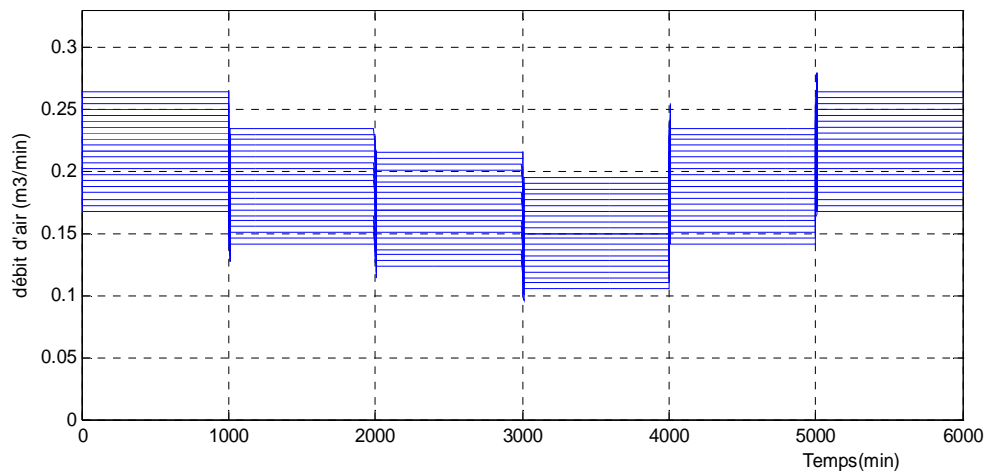


Figure IV.13- Intervalle de commande pour une incertitude de 1%

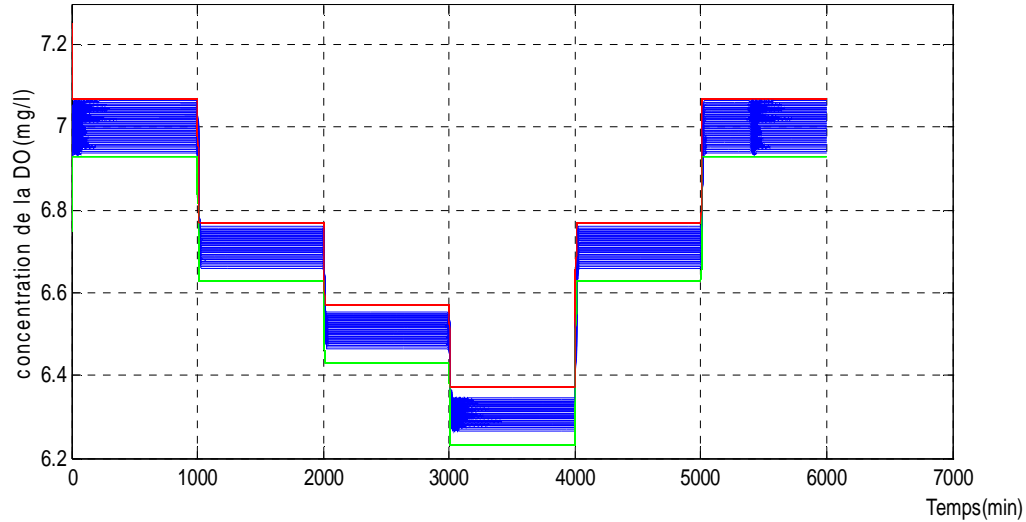


Figure IV.14-Sortie du système avec une incertitude de 1%

Clairement, le système en boucle fermé est plus robuste en terme de précision.

IV.3. Contrôle de la concentration du produit dans un réacteur chimique : CSTR

IV.3.1. Présentation et modèle du procédé :

Un des réacteurs chimiques les plus utilisés dans l'industrie est le réacteur ouvert complètement agité (CSTR) « Continuously Stirred Tank Reactor). Un composant A rentre dans le réacteur avec une concentration C_{a0} , une température T_f , et un débit q (signal de commande). A la sortie du réacteur la concentration du produit sera C_a et la température sera T_a . La réaction qui se déroule dans le réacteur CSTR est parfaitement agité est exothermique irréversible : $A \rightarrow B$. Le processus est modélisé par les équations différentielles (IV.21) et (IV.22) :

$$\dot{C}_a = \frac{q}{V}(C_{a0} - C_a) - a_0 C_a e^{\frac{-E}{RT_a}} \quad (IV.21)$$

$$\dot{T}_a = \frac{q}{V}(T_f - T_a) + a_1 C_a e^{\frac{-E}{RT_a}} + a_3 q_c [1 - e^{\frac{-a_2}{q_c}}] (T_{cf} - T_a) \quad (IV.22)$$

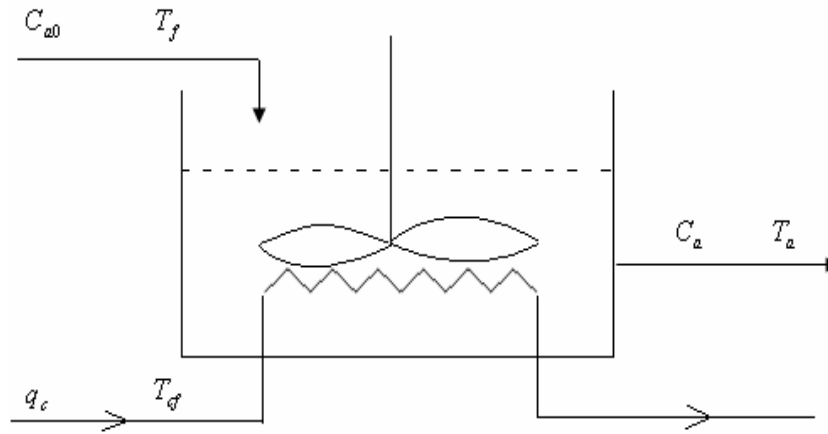


Figure IV.15 –le CSTR

Les conditions nominales pour la production de la concentration $C_a = 0.1 \text{ mol/l}$ sont : $T=438.54\text{k}$, $q_c = 103.41\text{L/min}$. On présente la signification des paramètres du CSTR et leurs valeurs nominales dans le tableau IV.5:

Paramètre	Signification	Valeur
Q	Débit du processus	100 mol/l
C_{a0}	Concentration du composant A	1 mol/l
T_f	Température d'alimentation	350 k
T_{cf}	Température du vapeur de refroidissement	350 k
V	Volume du réacteur	100 l
h_a	Coefficient du transfert de chaleur	7.10^5 J/min k
a_0	Facteur pré exponentiel	$7,2.10^{10} \text{ min}^{-1}$
E/R	Rapport	2.10^4 cal/mol
a_1	Constant	$1,44.10^{13}$
a_2	Constant	$6,987.10^2$
a_3	Constant	0.01

Tableau IV.5 - Paramètres du CSTR

IV.3.2. Commande par logique floue du réacteur :

IV.3.2.1. Le contrôleur flou

Le contrôleur flou de Mamdani est basé sur un ensemble de règles du genre Si- Alors:

Si antécédent **Alors** conséquence (linguistique).

Les entrées de notre contrôleur flou sont :

e : erreur entre la sortie du système et la référence ($y(i)-ref$).

de : différence entre l'erreur actuelle et l'erreur précédente ($e(i)-e(i-1)$).

Notre modèle flou est composé de neuf règles :

Règle 1 : Si (e est *négatif*) et (de est *négatif*) ; alors (q est *faible*)

Règle 2 : Si (e est *négatif*) et (de est *zéro*) ; alors (q est *faible*)

Règle 3 : Si (e est *négatif*) et (de est *positif*) ; alors (q est *moyen*)

Règle 4 : Si (e est *zéro*) et (de est *négatif*) ; alors (q est *faible*)

Règle 5 : Si (e est *zéro*) et (de est *zéro*) ; alors (q est *moyen*)

Règle 6 : Si (e est *zéro*) et (de est *positif*) ; alors (q est *élevé*)

Règle 7 : Si (e est *positif*) et (de est *négatif*) ; alors (q est *moyen*)

Règle 8 : Si (e est *positif*) et (de est *zéro*) ; alors (q est *élevé*)

Règle 9 : Si (e est *positif*) et (de est *positif*) ; alors (q est *élevé*) ;

L'univers de discours et la fonction d'appartenance de chaque variable sont données par :

La commande q :

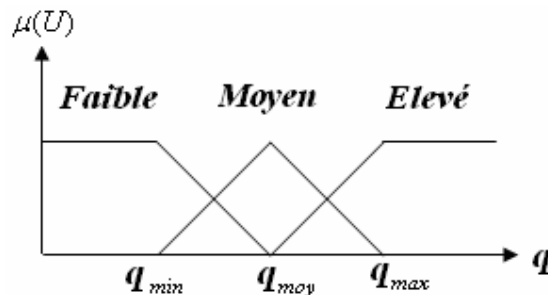


Figure IV.16- Fonction d'appartenance de la commande q

L'erreur e :

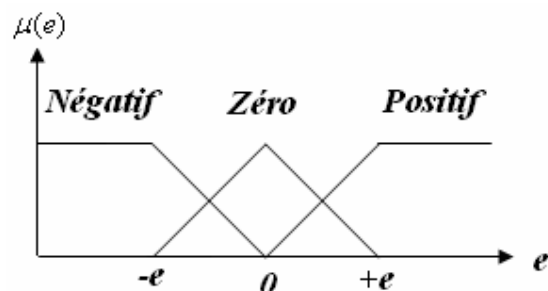


Figure IV.17 - Fonction d'appartenance de l'erreur e

Variation d'erreur :

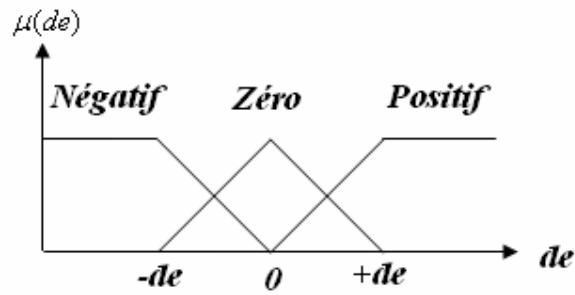


Figure IV.18 -Fonction d'appartenance de la déviance d'erreur e

IV.3.2.2. Résultats obtenus par le contrôleur flou:

Les résultats obtenus par le contrôleur flou sont donnés par les figures IV-19 :

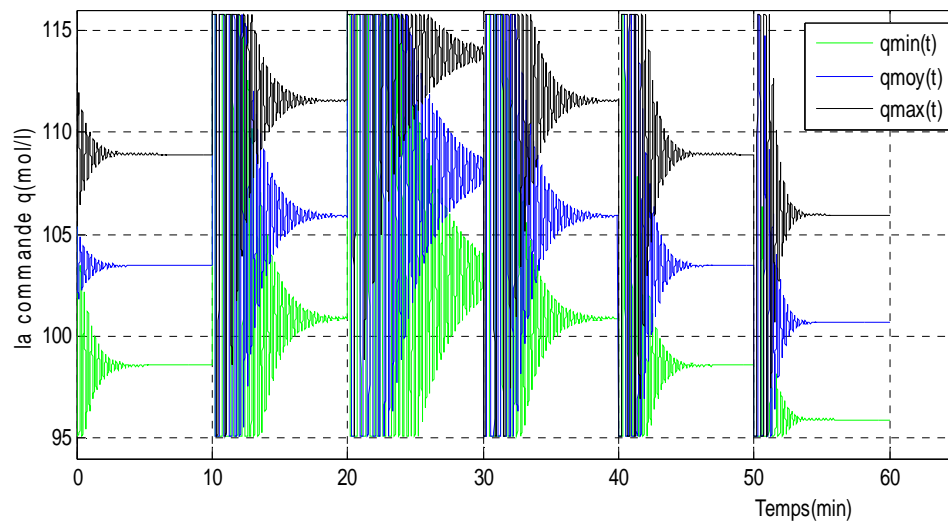


Figure IV.19 - L'intervalle du signal de commande délivré par le contrôleur flou

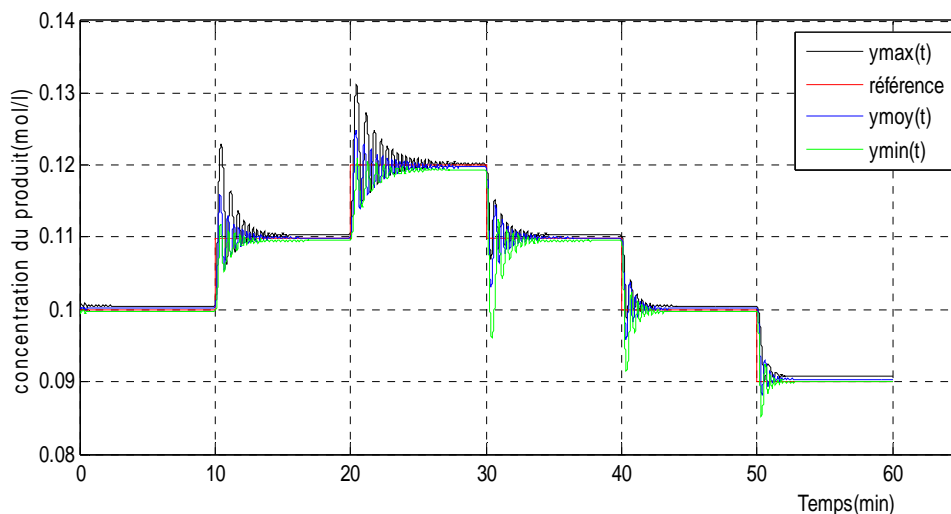


Figure IV.20 -L'intervalle de la sortie du système par la commande flou

Il est très difficile de trouver un contrôleur flou par expertise.

Après plusieurs tests, les résultats du meilleur contrôleur obtenus sont donnés dans les figures IV.19 et IV.20. La commande est très active et le régime de transition est très oscillant dans tous les cas.

IV.3.3. Contrôle par intervalle flou de la concentration du produit dans un réacteur chimique (CSTR) :

IV.3.3.1. Le modèle à intervalles flous

Le réacteur chimique CSTR est un système non linéaire deuxième ordre, la forme du modèle mathématique choisie pour ce procédé est le suivant :

$$ym(i+1) = a * ym(i) + b * ym(i-1) + c * u(i) + d \quad (IV.23)$$

Dont :

$$a \in [a_m, a_p] = [(a)^-, (a)^+] \quad ; \quad b \in [b_m, b_p] = [(b)^-, (b)^+] \quad ; \quad c \in [c_m, c_p] = [(c)^-, (c)^+]$$

$$d \in [d_m, d_p] = [(d)^-, (d)^+]$$

Pour déterminer les valeurs de $a_m, a_p, b_m, b_p, c_m, c_p, d_m, d_p$ on a utilisé le même algorithme génétique que précédemment. Les valeurs obtenues, pour $\alpha=0$, sont données dans le tableau suivant :

paramètre	valeur
a_p	1.9966

a_m	1.9966
b_p	0.9966
b_m	0.9966
c_p	4.7280e-008
c_m	2.8742e-008
d_p	3.5e-006
d_m	1.7153e-006

Tableau IV.6-Paramètres du modèle du procédé2

Les fonctions d'appartenance des paramètres sont les mêmes que l'exemple précédent. La figure IV-21 présente la sortie du système et du modèle pour les paramètres nominaux du système:

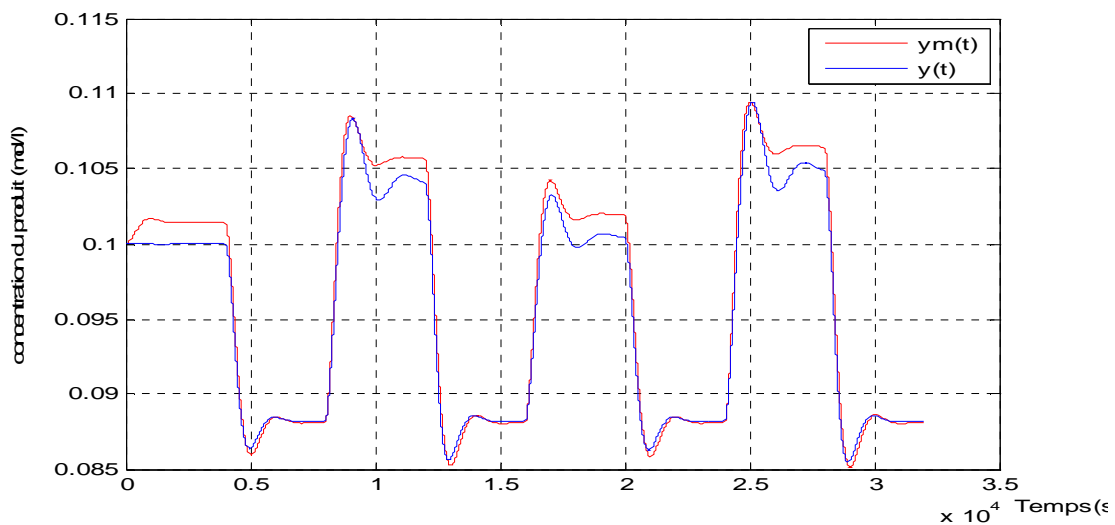


Figure IV.21-Sortie du système et du modèle

IV.3.3.2. Calcul et application de la commande :

Dans cette partie, le but est de maintenir, par *régulation*, la sortie du système dans une valeur fixe ($y_d=0.1$).

1°) Cas nominal

Par l'utilisation des formules de chapitre II, on peut facilement tirer u_m et u_p . La commande appliquée au système subit la même loi de commande de l'exemple précédent.

La figure IV-22 présente $y_d(t)$ et $y(t)$ sans aucune incertitude sur les paramètres nominaux du système :

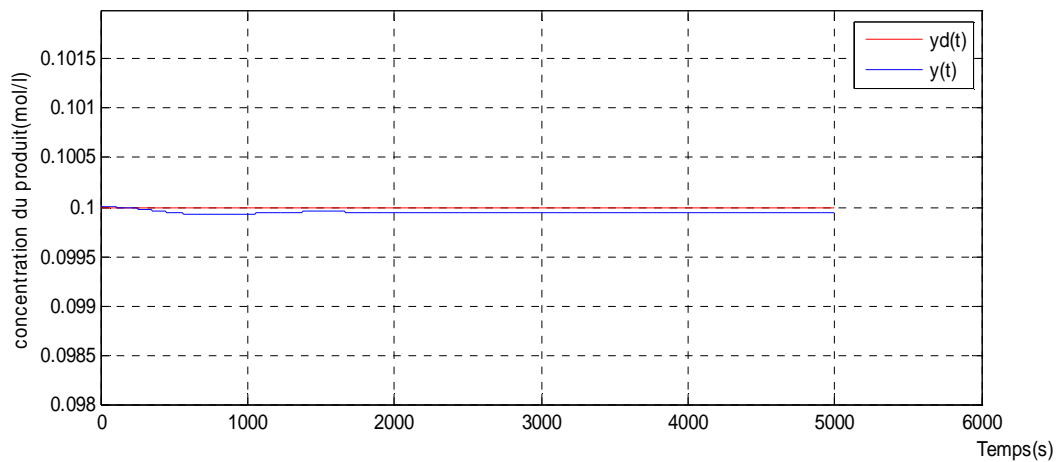


Figure IV.22-Sortie du système et référence

On remarque, par comparaison de la figure IV.20 avec la figure IV.22, que le système est très robuste en terme de précision et régime transitoire non oscillant.

2°) Cas avec incertitudes

On introduit des incertitudes sur les paramètres a_2 , a_3 telles que données dans le tableau IV 6.

Incertitude sur y_d	Intervalle de a_2	Intervalle de a_3
10%	[640,760]	[0.0097,0.01034]
5%	[650,710]	[0.00979, 0.0101]
3%	[670, 702]	[0.00985, 0.010015]

Tableau IV.6– Intervalles des paramètres du système

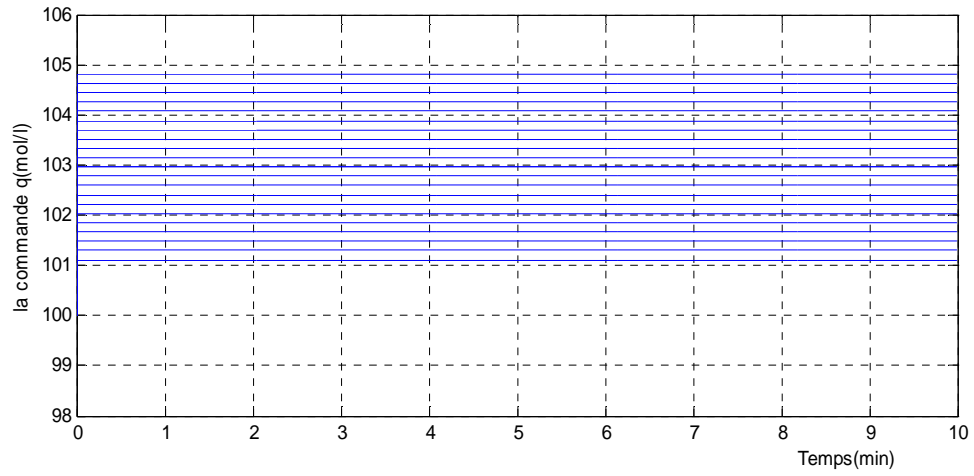


Figure IV.23-Intervalle de commande pour une incertitude de 10%

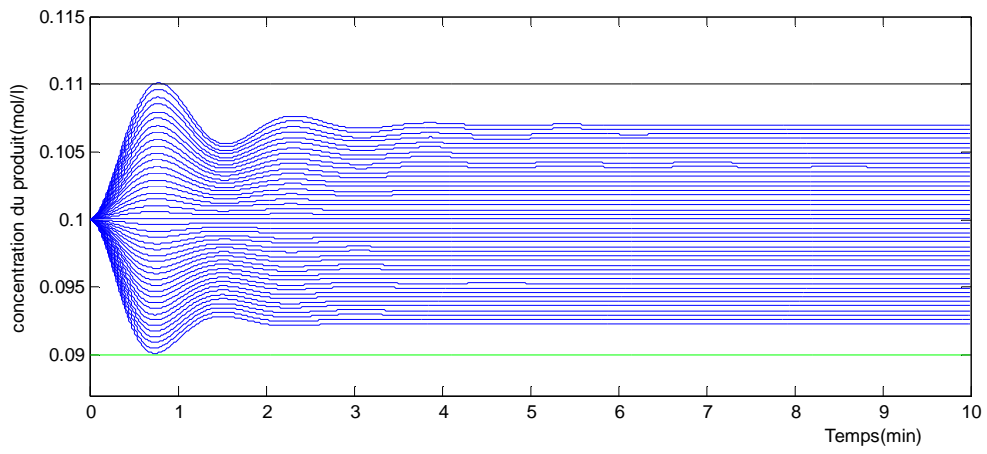


Figure IV.24-sortie du système avec une incertitude sur les paramètres de 10%

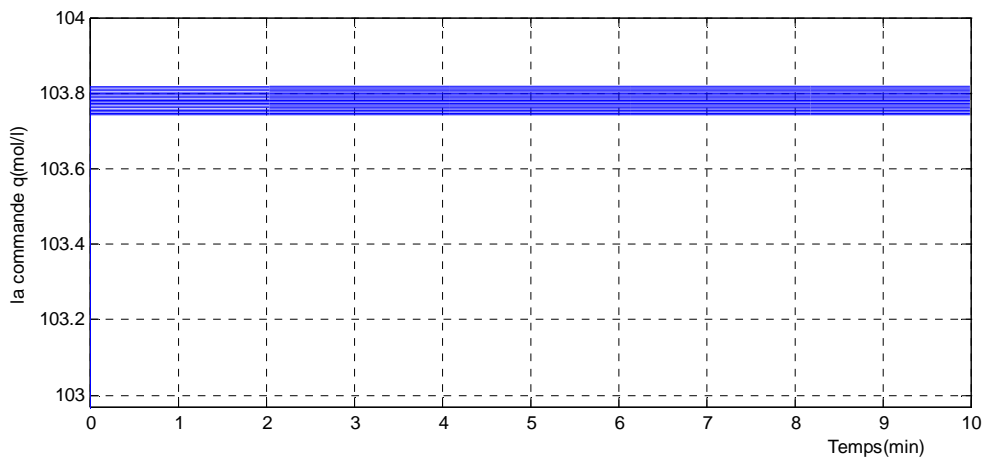


Figure IV.25-intervalle de commande pour une incertitude de 3%

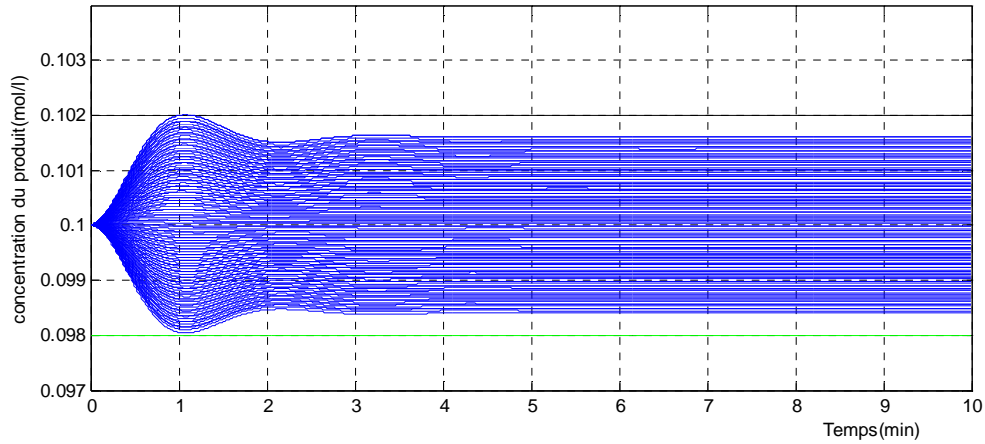


Figure IV.26-sortie du système avec une incertitude sur les paramètres de 3%

Par cette méthode, pas d'oscillation sur la commande, très bonne précision, une légère oscillation sur la sortie du système dans le domaine autorisé. Donc le système en boucle fermé est plus robuste en terme de précision.

Chapitre V :

Conclusion générale

Dans ce travail nous avons considéré l'application de la théorie des intervalles flous pour l'identification et la commande des systèmes non linéaires. Après avoir introduit les éléments de base de la théorie, nous avons accompli une étude détaillée dans deux cas particuliers. Dans les deux cas nous avons d'abord appliqué une commande par la logique floue classique. Ensuite en utilisant l'optimisation par les algorithmes génétiques, nous avons construit des modèles à intervalles flous pour les deux systèmes considérés en l'occurrence le système de purification des eaux usées et le réacteur continu. A partir de ces modèles, nous avons obtenus la commande par la méthode dite inverse. La commande finale à appliquer au système est alors construite par interpolation. La méthode fournit une approche systématique pour concevoir le contrôleur inverse ce qui n'est pas le cas du contrôleur flou qui est obtenu par expertise. De plus les différents tests de simulation ont montré que les systèmes obtenus sont robustes en ce sens qu'ils satisfont les critères de performances dans les limites préétablies.

Liste des figures

Figure II.1-Classification des températures d'une pièce en deux ensembles.....	4
Figure II.2- Schéma d'une commande floue.....	8
Figure IV.1-Schémas synoptique d'une station d'épuration des eaux.....	17
Figure IV.2-Schémas du procédé à boues activées.....	20
Figure IV.3-Signal de commande délivré par le contrôleur flou.....	23
Figure IV.4-Sortie du système et référence.....	24
Figure IV.5- L'intervalle du signal de commande délivré par le contrôleur flou.....	24
Figure IV.6 -L'intervalle de la sortie du système par la commande floue.....	25
Figure IV.7-Schéma de base de l'identification.....	25
Figure IV.8- Les intervalles flous.....	26
Figure IV.9-Sortie du système et du modèle.....	27
Figure IV.10-Sortie du système et référence.....	28
Figure IV.11- Intervalle de commande pour une incertitude de 10%.....	29
Figure IV.12-Sortie du système avec une incertitude sur les paramètres de 10%.....	29
Figure IV.13- Intervalle de commande pour une incertitude de 1%.....	29
Figure IV.14-Sortie du système avec une incertitude.....	30
Figure IV.15 –le CSTR.....	31
Figure IV.16- Fonction d'appartenance de la commande q	32
Figure IV.17 - Fonction d'appartenance de l'erreur e	32
Figure IV.18 -Fonction d'appartenance de la déférence d'erreur e	33
Figure IV.19 - L'intervalle du signal de commande délivré par le contrôleur flou.....	33
Figure IV.20 -L'intervalle de la sortie du système par la commande floue.....	34
Figure IV.21-Sortie du système et du modèle.....	35
Figure IV.22-Sortie du système et référence.....	36
Figure IV.23-Intervalle de commande pour une incertitude de 10%.....	36
Figure IV.24-sortie du système avec une incertitude sur les paramètres de 10%.....	37
Figure IV.25-intervalle de commande pour une incertitude de 2%.....	37
Figure IV.26-sortie du système avec une incertitude sur les paramètres de 2%.....	38

Liste des tableaux

Tableau IV.1-paramètres du réacteur.....	21
Tableau IV.2- base de règles.....	23
Tableau IV.3- Paramètres du modèle du procédé1.....	26
Tableau IV.4 – Intervalles des paramètres du système.....	28
Tableau IV.5 - Paramètres du CSTR.....	31
Tableau IV.6-Paramètres du modèle du procédé2.....	36
Tableau IV.7– Intervalles des paramètres du système.....	36

Bibliographie

- [1] K. Zhou, J.C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice Hall, New Jersey, 1995.
- [2] K.M. Passino and S. Yurkovich, *Fuzzy Control*, Reading, MA:Addison-Wesley, 1998.
- [3] D. Dubois, H. Prade, “Random sets and fuzzy interval analysis,” *Fuzzy Sets and Systems*, 42, pp.87–101, 1991.
- [4] S.-G. Li, “Fuzzy interval”, *Appl. Math. Lett.*, 14, pp. 737–740, 2001.
- [5] R. Moore and W. Lodwick, “Interval analysis and fuzzy set theory,” *Fuzzy Sets Syst.*, 135, pp. 5–9, 2003.
- [6] C. W. Kung, “Fuzzy interval control of mobile robots,” *Comput. Elect. Eng.*, 22, pp. 211–229, 1996.
- [7] Y.Wang, Q. L. Zhang, X. D. Liu, and S. C. Tong, “Stability analysis and design of fuzzy control systems based on interval approach,” in *Proc. 4th World Congr. Intelligent Control and Automation*, vol. 1, pp. 376–380, 2002.
- [8] Igor Skrjanc, Saso Blazic, Osvaldo Agamennoni. “Identification Of dynamical systems with a robust interval fuzzy model”, *Automatica*, 41, pp.327-332, 2005.
- [9] Reda Boukezzoula, L. Foulloy & S. Galichet “Inverse controller design for fuzzy interval systems”, *IEEE Transaction on Fuzzy Systems*,14, pp.111-123, 2006.

ANNEXE

Sur les algorithmes génétiques

1. Introduction

Dans les années soixante, John Holland étudie les systèmes évolutifs, et en 1975, il introduit le premier modèle formel des algorithmes génétiques (the canonical genetic algorithm AGC) dans son livre *Adaptation in Natural and Artificial Systems*. Il expliqua comment ajouter de l'intelligence dans un programme informatique avec les croisements (échangeant le matériel génétique) et la mutation (source de la diversité génétique). Ce modèle servira de base aux recherches ultérieures et sera plus particulièrement repris par Goldberg qui publiera en 1989, un ouvrage de vulgarisation des algorithmes génétiques.

2. Principe de fonctionnement des algorithmes génétiques :

Les algorithmes génétiques fournissent des solutions aux problèmes n'ayant pas de solutions calculables en temps raisonnable de façon analytique ou algorithmique.

Leur fonctionnement s'inspire de l'évolution génétique des espèces, ils copient de façon extrêmement simplifier certains comportements des populations naturelles. Ainsi, ces techniques reposent sur l'évolution d'une population de solution qui sous l'action de règles précises optimise un comportement donné exprimé sous forme d'une fonction, dite *fonction sélective (fitness function)*.

3. mécanisme des algorithmes génétiques :

Le schéma général ci-dessous décrit le fonctionnement d'un algorithme génétique :

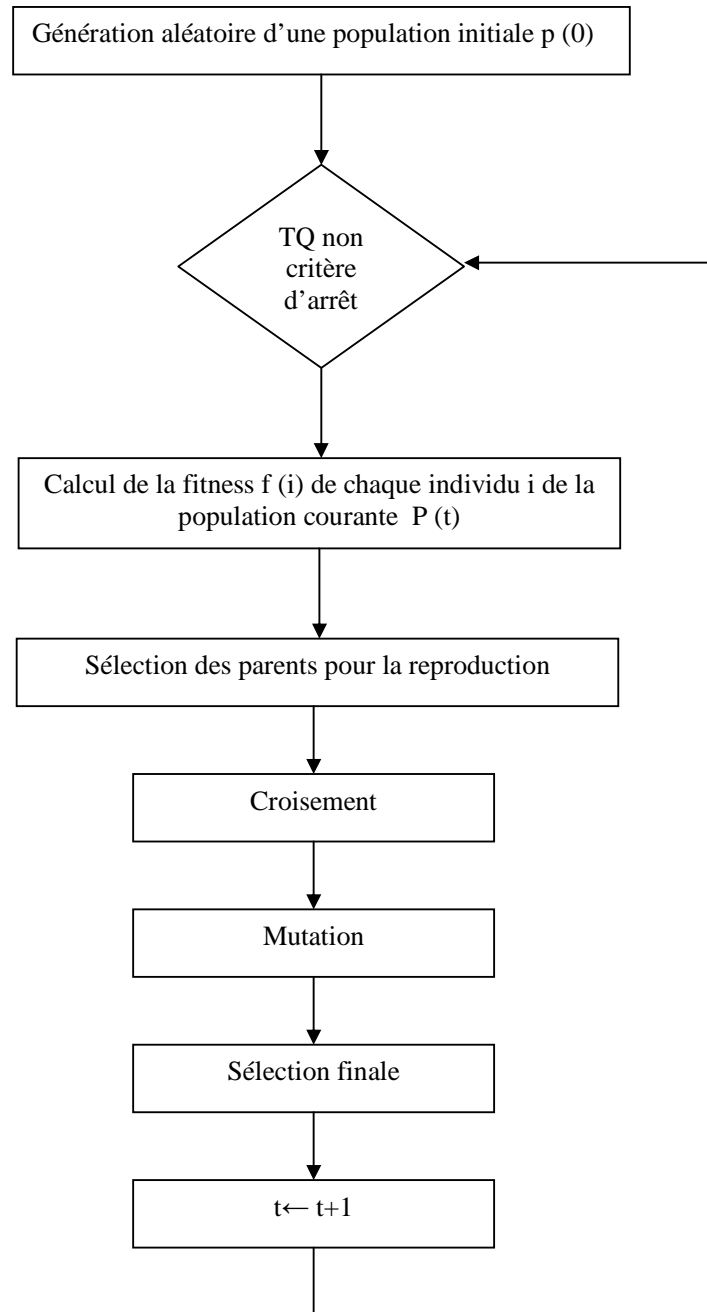


Schéma 1 : fonctionnement général d'un algorithme génétique

Un algorithme génétique est un algorithme itératif de recherche d'optimum, il manipule une population (figure1) de taille constante. Cette population est formée de points appelés

chromosomes, chaque chromosome représente le codage d'une solution potentielle au problème à résoudre et il est constitué d'un ensemble d'éléments appelés *gènes*.

A chaque itération, appelée génération, est créé une nouvelle population avec le même nombre de chromosomes. Cette génération consiste en des chromosomes mieux *adaptés* à leur environnement tel qu'il est représenté par la fonction sélective.

Au fur et à mesure des générations, les chromosomes vont tendre vers l'optimum de la fonction sélective. La création d'une nouvelle population à partir de la précédente se fait par l'application des *opérateurs génétiques* qui sont : *la sélection, le croisement et la mutation*.

La *sélection des meilleurs chromosomes*, est la première opération dans un algorithme génétique. Au cours de cette opération, l'algorithme sélectionne les éléments pertinents qui optimisent mieux la fonction. Le *croisement* permet de générer deux chromosomes nouveaux « enfant » à partir de deux chromosomes « parents », tandis que la *mutation* réalise l'inversion d'un ou plusieurs gènes d'un chromosome.

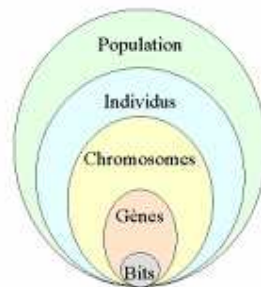


Figure-1- principaux éléments d'une population

4. Codage

Lors de la génération d'une population initiale, tous les individus doivent être codés selon une représentation spécifique. Le choix du codage à adopter est fonction des caractéristiques du présent problème. Il y a trois principaux types de codage utilisables, et on peut passer de l'un à l'autre facilement.

4.1. Codage binaire

HOLLAND a proposé de représenter les chromosomes comme étant des chaînes de bit, appelées chaînes binaires. Il a montré que cette méthode de codage est en principe la plus efficace, car elle fait intervenir un alphabet comportant uniquement deux symboles. Cependant, le codage binaire présente un inconvénient majeur. Lorsque le nombre de

paramètres à coder est important et que la précision souhaitée sur la solution est élevée, la chaîne binaire devient très longue.

4.2. Codage en nombres réels

En terme d'occupation mémoire, ce codage est moins intéressant puisqu'il nécessite autant de bits en mémoire pour représenter les variables. Par rapport au codage binaire, la représentation en nombre réels est mieux adaptée aux problèmes imposant une précision.

Dans un algorithme génétique utilisant un codage en nombre réels, chaque chromosome est en fait un vecteur dont les coordonnées sont les variables du processus d'optimisation.

4.3. Codage en base N

Certains auteurs ont proposé une autre méthode de codage appelée codage en base n, et ont décrit les opérateurs génétiques associés. Dans cette méthode, les éléments constituant un chromosome sont des chiffres exprimés dans une base de numérotation n. Ce qui permet de représenter n valeurs discrètes. Les opérateurs de croisement et de mutation adaptés au codage en base n sont de simples extensions des opérateurs standard. Pour le croisement, les chromosomes sont scindés puis recombinaés entre deux éléments codés en base n. Pour la mutation, l'élément initial est remplacé par un chiffre en base n tiré au sort

5. Opérateurs génétiques

Les opérateurs génétiques jouent un rôle prépondérant dans le possible réussite d'un algorithme génétique. Nous en dénombrons trois principaux : l'opérateur de *sélection*, de *croisement* et de *mutation*.

Si le principe de chacun de ces opérateurs est facilement compréhensible, il est toutefois difficile d'expliquer l'importance isolée de chacun de ces opérateurs dans la réussite de l'algorithme génétique. Cela tient en partie, au fait que chacun de ces opérateurs agit selon divers critères qui lui sont propres (valeur sélective des individus, probabilité d'activation de l'opérateur, etc....).

5-1. Opérateur de sélection

Cet opérateur peut-être le plus important, puisque il permet aux individus d'une population de survivre, de se reproduire ou de mourir. En règle générale, la probabilité de survie d'un individu sera directement reliée à son efficacité relative au sein de la population.

La probabilité de sélection d'un individu C_i est :

$$P_{\text{selection}}(C_i) = \frac{F_{\text{evaluation}}(C_i)}{\sum_{j=1}^n F_{\text{evaluation}}(C_j)}$$

n : la taille de la population de l'algorithme génétique.

$P_{\text{selection}}(C_i)$: probabilité de sélection de l'individu i .

$F_{\text{evaluation}}(C_i)$: la valeur d'évaluation de l'individu i .

Il existe plusieurs méthodes de sélection, la plus connue est *la roue de loterie biaisée (roulette wheel)* de Goldberg (1989). Selon cette méthode, chaque chromosome sera dupliqué dans une nouvelle population proportionnellement à sa valeur d'adaptation. On effectue, en quelque sorte, autant de tirages avec remise qu'il y a d'éléments dans la population.

L'inconvénient majeur de cette méthode, repose sur le fait qu'un individu n'étant pas le meilleur peut tout de même dominer la sélection. Elle peut aussi engendrer une perte de diversité par la domination d'un super individu.

Il existe d'autres méthodes de sélection, les plus connues sont :

- a. *Sélection par élitisme* : Les individus sont triés selon leur fonction d'adaptation. Seule la moitié supérieure de la population correspondant aux meilleurs composants est sélectionnée.
- b. *Sélection par tournoi* : Cette technique consiste à choisir aléatoirement deux individus et à comparer leur fonction d'adaptation. On accepte la meilleure solution pour accéder à la génération intermédiaire. On répète cette opération jusqu'à remplir la génération intermédiaire ($N/2$ composants). Les individus qui gagnent à chaque fois peuvent être copiés plusieurs fois. Cela favorise la pérennité de leurs gènes.

5.2. Opérateur de croisement

L'opérateur de croisement utilisé par les algorithmes génétiques est la transposition informatique du mécanisme qui permet, dans la nature, la production de chromosomes qui héritent partiellement des caractéristiques des parents.

Tout d'abord, deux individus, qui forment alors un couple, sont tirés au sein de la nouvelle population issue de la reproduction. Puis un site de croisement est choisi aléatoirement.

Enfin, selon une probabilité P_c que le croisement s'effectue, les segments finaux (dans le cas d'un seul site de croisement) des deux parents sont alors échangés autour de ce site.

Cet opérateur permet la création de deux nouveaux individus.

Un individu sélectionné lors de la reproduction ne subit pas nécessairement l'action d'un croisement. Le croisement ne s'effectue qu'avec une certaine probabilité.

Plus cette probabilité est élevée, plus la population subira de changement.

Il existe différents types de croisement :

- Croisement selon un point : détermine aléatoirement un point de coupure et échange la deuxième partie des deux parents.

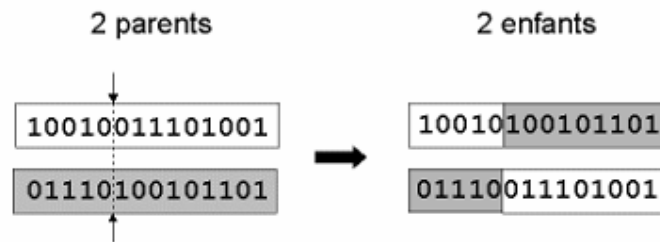


Figure- croisement selon un point

- Croisement selon deux points (peut être étendu à x points) : possède deux points de coupure qui sont déterminés aléatoirement.



Figure- croisement selon deux points

- Croisement uniforme : échange chaque bit avec une probabilité fixé à 0,5.

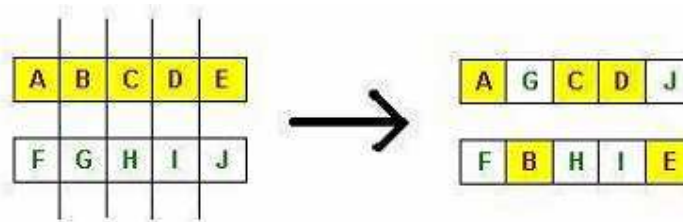


Figure- croisement uniforme

5.3. Opérateur de mutation

En biologie, une mutation est une modification d'un chromosome qui n'est pas issu d'une opération de croisement. Il est assez difficile de cerner l'effet d'une mutation dans le cas des organismes vivants. D'après les expérimentations réalisées, la mutation évite qu'une

population soit stagnée lorsque tous ses individus soient génétiquement identiques. En d'autres termes, l'opération de mutation permet de faire apparaître de nouveaux gènes [18].

La mutation originale est celle appliquée à des chromosomes codés en binaire. Chaque bit est remplacé selon la probabilité pm par son inverse. On génère un nombre aléatoire entre (0 et 1), si ce nombre est supérieur ou égale à pm alors :

Si la valeur du bit est égale à 0 alors elle sera remplacée par 1 et si elle est égale à 1 alors elle sera remplacée par 0.

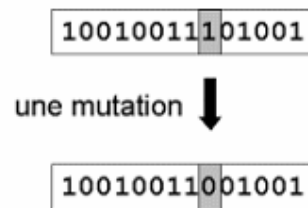


Figure- une mutation

6. Sélection finale

Cette dernière étape du processus itératif consiste en l'incorporation des nouvelles solutions dans la population courante. Les nouvelles solutions sont ajoutées à la population courante en remplaçant totalement ou partiellement les anciennes solutions. Généralement, les meilleures solutions remplacent les plus mauvaises. Il en résulte une amélioration de la population.

7. Autres paramètres

Les opérateurs génétiques sont guidés par un certain nombre de paramètres fixés à l'avance.

La valeur de ces paramètres influe sur la réussite ou non d'un algorithme génétique. Ces paramètres sont les suivants :

- *La taille de la population N* : Si N est trop grand, le temps de calcul de l'algorithme peut s'avérer très important, et s'il est trop petit, il peut converger trop rapidement vers un mauvais chromosome. Cette importance de la taille est essentiellement dû à la notion de parallélisme implicite qui implique que le nombre d'individus traité par l'algorithme est au moins proportionnelle au cube du nombre d'individus.

- *La probabilité de croisement P_c* : elle dépend de la forme de la fonction de fitness. Son choix est généralement heuristique. Plus elle est élevée, plus la population subit de changements importants.
- *La probabilité de mutation P_m* : ce taux est généralement faible, puisqu'un taux élevé risque de conduire à une solution sous-optimale. Plutôt que réduire P_m , une autre façon d'éviter que les meilleurs individus soient altérés, est d'utiliser la reconduite explicite de l'élite dans une certaine proportion. Ainsi, bien souvent, les meilleurs 5% par exemple, de la population sont directement reproduits à l'identique, l'opérateur de reproduction ne jouant alors que sur les 95% restant. Cela est appelé *stratégie élitiste*.

Résumé

Dans ce travail, on a étudié une comparaison entre deux contrôleurs : contrôleur par intervalle flou et contrôleur par logique flou classique.

Pour la première approche, un algorithme génétique est utilisé pour l'identification des systèmes à partir des données d'entrée- sortie, ensuite l'intervalle de la commande est déterminé par le modèle et la référence . La commande appliquée varie selon la variation des paramètres du système. Les résultats obtenus par cette méthode de contrôle sont encourageants lorsque comparés avec celle obtenus par le contrôle du même procédé avec un contrôleur flou classique. Cette étude est effectuée sur deux systèmes non linéaire : un procédé de traitement des eaux par boues activés et le réacteur CSTR.

Abstract

In this work we have studied a comparison between two controllers: a fuzzy interval controller and a classical fuzzy logic controller.

For the first approach a genetic algorithm is used for the identification of the systems from input-output data, and then the interval of the control signal is determined by the model and the reference. The applied command varies according to the variation of system parameters. Results obtained by this method of control are encouraging when compared with those obtained by the control of the same system with a classic fuzzy controller. This study is done on two non linear systems: water treatment by activated muds process and the CSTR reactor.

ملخص

في هذا العمل قمنا بدراسة مقارنة بين نظامي تحكم: الأول هو نظام تحكم بالمجال المبهم، والثاني هو نظام تحكم بالمنطق المبهم الكلاسيكي.

في المقاربة الأولى، استخدمنا خوارزم وراثي للتعريف بالأنظمة عن طريق المعلومات (دخول-خروج) . بعد ذلك مجال التحكم معرف عن طريق نمط النظام و إشارة المرجع.

قيمة إشارة التحكم المطبقة على النظام تتغير بتغير معالم النظام. النتائج المحصل عليها بطريقة التحكم هاته مشجعة إذا ما قورنت مع تلك المحصل عليها بطريقة التحكم بالمنطق المبهم الكلاسيكي.

هذه الدراسة أجريت على نظامين غير خطيين: الأول هو نظام معالجة المياه الملوثة بالأحوال المتحركة، والثاني هو مفاعل كيميائي مفتوح.