

II.1 Introduction

La logique floue est une branche des mathématiques et, à ce titre, toute une série de notions fondamentales sont développées. Ces notions permettent de justifier et de démontrer certains principes de base. Dans ce qui suit, on ne retiendra que les éléments indispensables à la compréhension du principe du réglage dans les systèmes électroniques par la logique floue.

Dans ce travail, nous allons présenter les principes de base de la logique floue, ainsi que ses domaines d'application. Ensuite, on donnera la description de la commande par logique floue avec ces différentes étapes de fuzzification, inférence et défuzzification et en fin les avantages et les inconvénients du réglage par la logique floue.

II.2 Les bases de la logique floue

II.2.1 Principe de la logique floue

Comparativement à la logique classique, les bases théoriques de la logique floue sont établies de manière à pouvoir traiter des variables inexactes de valeurs comprises entre 0 et 1, par contre la logique de Boole dont les variables ne peuvent prendre que les valeurs 0 et 1.

A titre d'exemple la classification des personnes à travers leur âge par les deux logiques présentées dans la figure (II.1). [6]

Fait apparaître que :

1. La logique classique (logique de Boole) n'admet pour les variables que les valeurs 0 et 1, qui font que les personnes âgées de moins de 30 ans sont systématiquement jeunes et les plus de 50 ans sont âgées, sans pour autant que, cette classification soit logique.
2. Alors que la logique floue, dont les variables peuvent prendre n'importe quelle valeur comprise entre 0 et 1, permet de tenir compte du passage progressif de l'individu d'un âge à un autre, on parle alors, de fonction d'appartenance μ .

Les limites ne varient pas soudainement, mais progressivement, la figure (II.1) montre une classification possible ; une personne de 25 ans appartient à l'ensemble (jeune) avec une valeur $\mu = 0.75$ de la fonction d'appartenance, et à l'ensemble (entre deux âges) avec $\mu = 0.25$. par contre une personne de 70 ans appartient avec une valeur $\mu = 1$ de la fonction d'appartenance à l'ensemble (âgé).

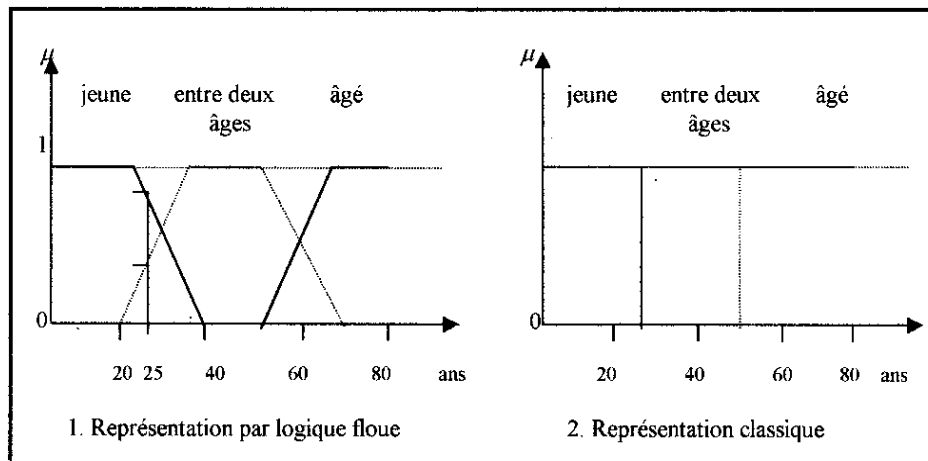


Figure II.1 : Classification selon la logique classique et selon la logique floue

II.2.2 Définition d'un sous-ensemble flou

Dans la théorie des ensembles classiques, il n'y a que deux situations acceptables pour un élément, appartenir ou ne pas appartenir à un sous-ensemble. Le mérite de Zadeh été de tenter de sortir de cette logique booléenne en introduisant la notion d'appartenance pondérée : permettre des graduations dans l'appartenance d'un élément à un sous-ensemble, c'est-à-dire d'autoriser un élément à appartenir plus moins fortement à ce sous-ensemble. Soit X un ensemble de référence et soit x un élément quelconque de X . Un sous ensemble flou A de X est défini comme l'ensemble des couples :

$$A = \{X, \mu_A(X), X \in X\}$$

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

Ainsi, un sous-ensemble flou A de X est caractérisé par une fonction d'appartenance $\mu_A(x)$ qui associe, à chaque point x de X un réel dans l'intervalle $[0, 1]$; $\mu_A(x)$ représente le degré d'appartenance de x à A . On observe les trois cas possibles suivants :

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

Où, $\mu_A(x) = 0$ si x n'appartient pas à A ; $0 < \mu_A(x) < 1$ si x appartient partiellement à A ; et $\mu_A(x) = 1$ si x appartient entièrement à A . La fonction d'appartenance $\mu_A(x)$ inclut ou exclut donc à ses extrémités, tout élément x au sous-ensemble A , mais entre les valeurs extrêmes le degré d'appartenance varie à proportion de la proximité à l'ensemble.

On peut faire remarquer que si A est un sous-ensemble classique, la fonction d'appartenance qui lui est associée ne peut prendre que les valeurs extrêmes 0 et 1.

On a dans ce cas :

$$\mu_A(x) = \begin{cases} 0, & \text{si } x \notin A \\ 1, & \text{si } x \in A \end{cases} \quad (\text{II.2})$$

II.2.3 Les variables linguistiques

Une variable linguistique sert à modéliser les connaissances imprécises sur une variable dont la valeur précise peut être inconnue [4][5].

Définition : Une variable linguistique est un triplet (x, U, T_x) dans lequel x est une variable définie sur un ensemble de référence U . L'ensemble $T_x = \{A_1, A_2, \dots\}$ fini ou infini, contient des sous-ensembles flous de U , utilisable pour caractériser x .

Exemple :

Soit la pression u interprétée comme une variable linguistique. Elle peut être décomposée à l'ensemble des termes suivants : $T(\text{pression}) = T_u = \{\text{faible}, \text{moyenne}, \text{OK}, \text{forte}, \text{énorme}\}$ où chaque terme T_u est caractérisé par un ensemble flou dans l'univers de discours $U = [-100\text{psi}, 100\text{psi}]$. Ces termes peuvent être caractérisés comme des ensembles flous pour lesquels les fonctions d'appartenance sont illustrées dans la figure II.2.

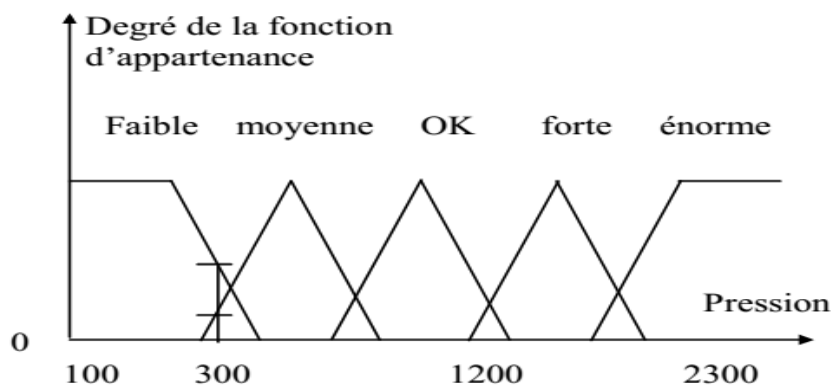


Figure II.2 : Fonctions d'appartenances de $T(\text{pression})$.

Les valeurs mesurées de la pression (x) se situent sur l'axe de pression. Dans cet exemple, une ligne verticale de n'importe quelle valeur mesurée rencontre au plus deux fonctions d'appartenances. Alors, par exemple, $x = 300$ appartient aux ensembles flous 'faible pression' et 'moyenne pression' à des degrés d'appartenances différents.

II.2.4 Les fonction d'appartenance

Le degré d'appartenance devient une fonction qui peut prendre une valeur réelle entre 0 et 1 avec un symbole plus utilise est « μ ». L'univers de discours est l'ensemble des valeurs réelles

que peut prendre la variable floue. Les formes les plus utilisées pour les fonctions d'appartenance est les formes trapézoïdales ou triangulaires. Ils s'agissent des formes les plus simples, composées par morceaux de droites. L'allure est complètement définie par trois points pour la forme triangulaire et quatre pour la forme trapézoïdales (Figure .II.3). Dans notre cas nous utiliserons la forme triangulaire [11, 14]. Cette forme est définie comme suit :

$$\mu(x) = \begin{cases} \frac{x-x_1}{x_2-x_1} & \text{si } x_1 \leq x \leq x_2 \\ \frac{x_3-x}{x_3-x_2} & \text{si } x_2 \leq x \leq x_3 \\ 0 & \text{si non} \end{cases} \quad (\text{II. 3})$$

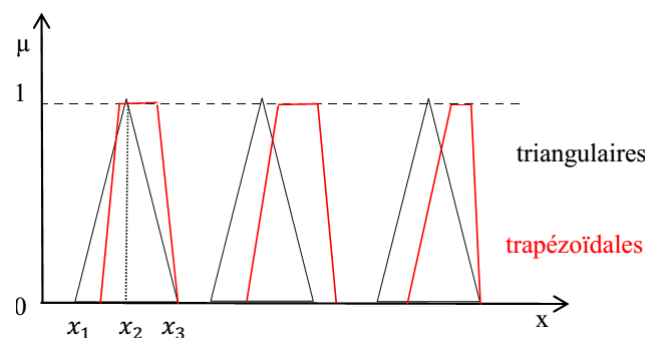


Figure. II.3 : Formes usuelles des fonctions d'appartenance

II.2.5 Opérateurs de la logique floue

La description d'une situation où il y a plus qu'une variable qui intervient, nécessite l'utilisation des opérateurs logiques tels que : "ET", "OU", et "NON".

Dans la théorie de la logique floue l'opérateur "ET" correspond à l'opération "Minimum", "OU" à l'opération "Maximum", et "NON" au complément à un.

L'analogie d'utilisation de ces opérateurs dans les deux logiques classiques et floue est sur le Tableau II.1

Tableau II.1 : Application des opérateurs dans les deux ensembles.

	Logique classique	Logique floue
C=A ET B	$C=A \cap B$	$\mu_C(x) = \min(\mu_A(x), \mu_B(x))$
C=A OU B	$C=A \cup B$	$\mu_C(x) = \max(\mu_A(x), \mu_B(x))$
C=NON A	$\bar{C} = \bar{A}$	$\mu_C(x) = 1 - \mu_A(x)$

Avec : A, B, C : ensembles

II.2.5.1 Opérateur ET (Intersection floue)

Le sous-ensemble flou, correspondant à l'intersection des sous-ensembles E et F est défini par les éléments x de l'univers de discours UD qui appartiennent à E et à F.

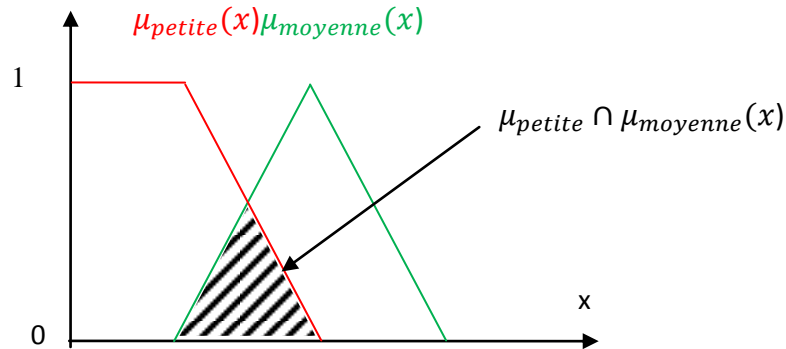


Figure II.4: Intersection des sous-ensembles flous « petit » et « moyenne » pour la variable linguistique (Taille).

Dans la logique floue, l'opérateur *ET* peut être exprimé par :

$$\mu_{E \cap F}(x) = \min\{\mu_E(x), \mu_F(x)\} \forall x \in UD \quad (\text{II. 4})$$

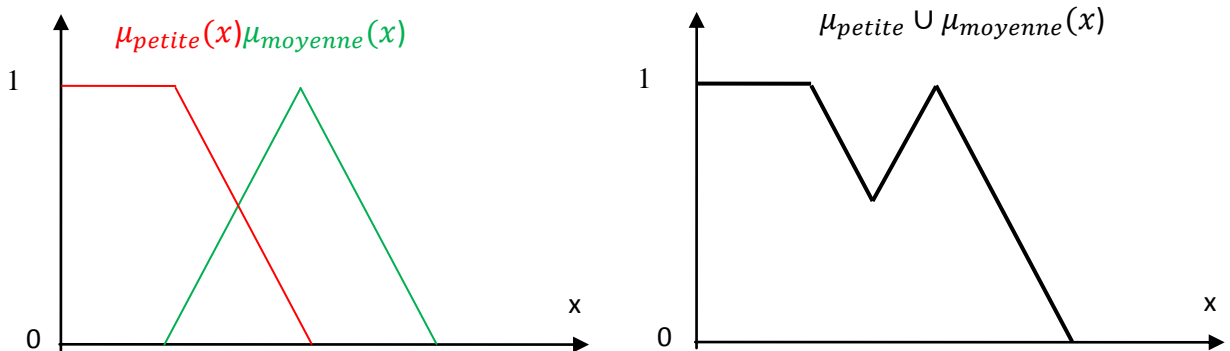
$$\mu_{E \cap F}(x) = \mu_E(x), \mu_F(x) \forall x \in UD \quad (\text{II.5})$$

II.2.5.2 Opérateur OU (Union floue)

Le sous-ensemble flou correspondant à l'union des sous-ensembles E et F est un sous-ensemble de l'univers de discours UD défini par tous les éléments x de UD qui appartiennent ou bien à E ou bien à F, ce que l'on note $(E \cup F)$. L'opérateur OU est généralement réalisé par la formation du maximum, quel'on exprime comme suit :

$$\mu_{E \cup F}(x) = \max\{\mu_E(x), \mu_F(x)\} \forall x \in UD \quad (\text{II. 6})$$

$$\mu_{E \cup F}(x) = \mu_E(x) + \mu_F(x) \forall x \in UD \quad (\text{II.7})$$



a) Partition floue de l'univers de discours

b) Ensemble flou : « TP ou TM »

Figure II.5 Union des sous-ensembles flous « petit » et « moyenne » pour la variable linguistique (Taille).

II.2.5.3 Operateur NON (complémentation floue)

Comme l'illustre la figure (II.6), le sous-ensemble flou complémentaire du sous-ensemble E est un sous-ensemble de l'univers de discours UD défini par les éléments x de l'UD qui n'appartiennent pas au sous-ensemble flou E. On peut exprimer ça par :

$$\mu_E(x) = 1 - \mu_E(x) \forall x \in UD \quad (II.8)$$

Le complément flou représente l'opération NON de la logique classique au sens flou.

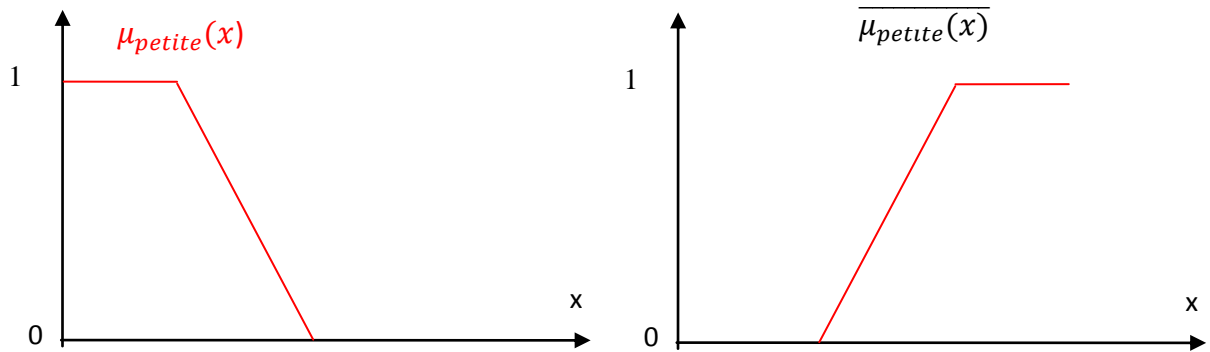


Figure II.6 : Complémentation de sous-ensemble flou « petite »

II.2.6 Règles d'inférence floue

Les systèmes basés sur la logique floue utilisent pour prendre des décisions la connaissance humaine présentée sous forme de règles floues, appelées aussi règles d'inférence. Elles sont exprimées sous la forme :

SI (prémisse) **ALORS** (conclusion)

Nous aurons par exemple :

Si (pression forte **ET** température élevée) **ALORS** (ouverture vanne grande)

Une règle floue est donc une combinaison entre une condition, nommée *prémisse* ou *prédicat* qui peut dépendre de plusieurs variables liées entre elles par des opérateurs ET, OU, NON et une *conclusion* ou *conséquence*. Les prémisses et conclusions forment des propositions floues exprimées par une conjonction ou une disjonction de prédicats, les conclusions sont obtenues par implication des propositions floues.

Ainsi en logique floue, on interprète la règle suivante : SI x est E ALORS y est F par le fait que si la variable floue x appartient au sous-ensemble E avec un degré d'appartenance $\mu_E(x)$, alors y appartient au sous-ensemble flou F avec un degré d'appartenance $\mu_F(x)$, qui dépend de la validité $\mu_E(x)$, de la prémisse. Plus généralement, l'expertise est donnée sous forme d'un ensemble de U règles, désigné par le terme de système d'inférence flou (SIF), présenté par une énumération du type :

SI [(prédicat 1) **ET/OU** (prédicat 1') **ET/OU**.....] **ALORS** (conclusion 1) **OU**

SI [(prédicat 2) **ET/OU** (prédicat 2') **ET/OU**.....] **ALORS** (conclusion 2) **OU**

SI [(prédicat Z) **ET/OU** (prédicat Z') **ET/OU**.....] **ALORS** (conclusion Z) **OU**

L'opérateur OU n'est pas utilisé dans les conclusions car il introduirait une incertitude dans la connaissance, l'expertise ne permettrait pas de déterminer quelle décision à prendre [7]. De même, l'opérateur NON n'est pas employé, en effet si une règle avait par exemple la conclusion : «*ALORS pression NON forte*», il serait impossible de dire si cela signifie «*pression faible*» ou «*pression moyenne*», cela serait encore un cas d'incertitude. Quatre étapes sont donc nécessaires pour obtenir la conclusion finale :

- Le calcul des propositions ;
- Le calcul des relations ;
- Les compositions des règles avec les faits observés ;
- L'agrégation des conclusions des règles.

II.3 Description d'une commande par la logique floue

La logique floue est principalement utilisée dans les domaines de prise de décision, de reconnaissance des formes, de modélisation et de commande des procédés. La commande ou la régulation des systèmes est le domaine industriel de la logique floue le plus exploité. On distingue trois structures majeures de régulateurs à logique floue (RLF) [8].

- La structure pure ;
- La structure de Takagi-Sugeno-Kang (TSK) ;
- La structure de Mamdani ou le modèle « fuzzification - defuzzification » ;

❖ La structure pure

Dans la structure pure les variables d'entrée et de sortie du RLF sont des variables floues ou linguistiques. Ceci constitue un handicap étant donné que les entrées et les sorties des régulateurs des systèmes réels sont des variables réelles ou numériques.

❖ La structure de Takagi-Sugeno-Kang (TSK)

La structure TSK résout ce problème par une simple transformation des variables linguistiques en variables réelles. L'inconvénient de cette structure est que le conséquent de chaque règle soit une formule mathématique.

Les règles de Takagi Sugeno

La première partie d'une règle de type Takagi –Sugeno est similaire à celle de Mamdani

tandis que la deuxième est une fonctionnelle.

La forme typique de cette règles 'écrit donc:

Si x_1 est E_1 (et) x_2 est E_2 (et) (et) x_m est E_m **Alors**

$$u_1 = f_1(x_1, \dots, x_m), u_2 = f_2(x_1, \dots, x_m), \dots, u_n = f_n(x_1, \dots, x_m),$$

Où f_1, \dots, f_n sont des fonctions réelles, théoriquement elles peuvent être linéaires ou non linéaires mais l'implémentation de la méthode exige qu'elles doivent être des fonctions linéaires.

❖ La structure de Mamdani

Mamdani s'est proposé une interface de défuzzification (défuzzificateur) à la sortie de la structure pure.

Le fuzzificateur transforme les variables réelles d'entrée en variables linguistiques floues, tandis que le défuzzificateur effectue l'opération inverse. La structure de Mamdani est devenue le modèle standard du RLF le plus utilisé dans la régulation des systèmes.

Les règles de Mamdani

La forme typique d'une règle Mamdani s'écrit :

Si x_1 est E_1 (et) x_2 est E_2 (et) (et) x_m est E_m **Alors**

$$u_1 \text{ est } S_1, u_2 = S_2, \dots, u_n \text{ est } S_n$$

Ou : x_1, \dots, x_m : les variables d'entrées

u_1, \dots, u_n est : les variables de sortie

$E_1, \dots, E_m, S_1, \dots, S_n$ **Alors** : ensembles flous

(Est): signifier l'appartenance

Comme l'illustre la figure (II.7), l'architecture de Mamdani est constituée de quatre parties essentielles à savoir :

- L'interface de fuzzification (le fuzzificateur) ;
- La base de connaissance ;
- Le mécanisme d'inférence ou l'évaluation de règles ;
- L'interface de défuzzification (le défuzzificateur).

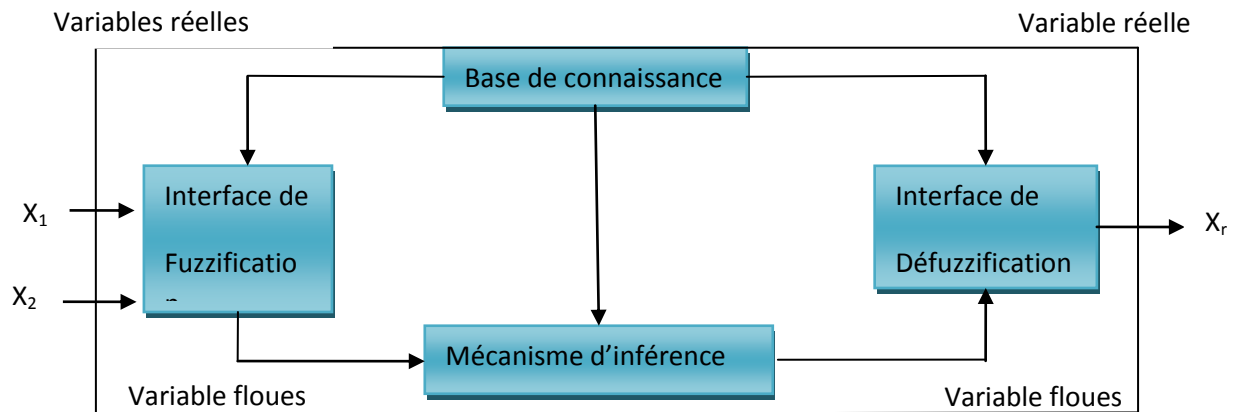


Figure : II.7 : Structure de base d'un contrôleur flou

Comme le système à commander ne reçoit que des valeurs déterministes (non-floues), un RLF devrait convertir les valeurs déterministes à son entrée en valeurs floues, les traiter avec les règles floues et reconvertir le signal de commande de valeurs floues en valeurs déterministes pour appliquer au procédé. Les rôles de chaque bloc peuvent être résumés dans les paragraphes suivant[8] :

II.3.1 Interface de fuzzification

Le bloc de fuzzification effectue les fonctions suivantes :

- ✚ Établit les plages de valeurs pour les fonctions d'appartenance à partir des valeurs des variables d'entrées ;
- ✚ Effectue une fonction de fuzzification qui convertit les données d'entrée en valeurs linguistiques convenables qui peuvent être considérées comme l'étiquette des ensembles flous. Cette opération doit être effectuée dans un domaine normalisé généralement par l'intervalle $[-1, 1]$ afin de faciliter le calcul.

II.3.2 Base de connaissance

Le bloc base de connaissance comporte une connaissance dans le domaine d'application et le résultat de commande prévu. Il consiste en « *base de données* » et en « *base de règles linguistiques (floues) de commande* » :

- ✚ La base de données effectue des définitions qui sont nécessaires pour établir les règles de commande et manipuler les données floues dans un RLF ;
- ✚ La base de règles représente la stratégie de commande et le but désiré par le biais des règles de commande linguistiques.

II.3.3 Mécanisme d'inférence

Le bloc inférence est le cœur d'un RLF, il possède la capacité de simuler les décisions humaines et de déduire les actions de commande floue à l'aide de l'implication floue et des règles d'inférence dans la logique floue. Le traitement numérique des règles d'inférence qui permet d'obtenir la sortie linguistique ou floue du régulateur se fait par différentes méthodes, on cite principalement [10] :

- La méthode d'inférence max-min ;
- La méthode d'inférence max-produit ;
- Et la méthode d'inférence somme-produit.

Chacune de ces trois méthodes utilise un traitement numérique propre des opérateurs de la logique floue :

- ✚ Pour la méthode d'inférence max-min, l'opérateur ET est réalisé par la formation du minimum, l'opérateur OU est réalisé par la formation du maximum, et ALORS, (l'implication) est réalisée par la formation du minimum.
- ✚ Pour la méthode d'inférence max-produit, l'opérateur ET est réalisé par la formation du produit, l'opérateur OU est réalisé par la formation du maximum, et ALORS (l'implication) est réalisée par la formation du produit.
- ✚ Pour la méthode d'inférence somme-produit, on réalise au niveau de la condition, l'opérateur ET par la formation de la somme (valeur moyenne), et l'opérateur ET par la formation du produit. Pour la conclusion, l'opérateur ALORS est réalisé par un produit.

Dans le cas de la méthode somme-produit, les actions des différentes règles sont liées entre elles par l'opérateur OU qui est réalisé par la formation de la moyenne arithmétique (somme moyenne). Cette méthode d'inférence est particulièrement avantageuse par rapport aux autres et nécessite une envergure de calcul relativement restreinte.

La fonction résultante dans ce cas peut être donnée comme suit [9] :

Si on a 2 variables d'entrées (x_1, x_2) , la fonction résultante de U règles pour la variable de sortie, x_r sera :

$$\mu_{RES}(x_r) = \frac{(\mu_{R1}(x_r) + \mu_{R2}(x_r) + \dots + \mu_{RZ}(x_r))}{Z} \quad (II.9)$$

Où

$$\mu_{Ri}(x_r) = \mu_i(x_1) \cdot \mu_i(x_2) \cdot \mu_{0i}(x_r) = \mu_{ci} \mu_{0i}(x_r); \quad i = 1, 2, \dots, z \quad (\text{II.10})$$

Avec

$\mu_{Ri}(x_r)$: Est la fonction d'appartenance partielle de chaque règle ;

μ_{ci} : Est le degré de vérification de la $i^{\text{ème}}$ règle ou condition ;

$\mu_{0i}(x_r)$: Est la fonction d'appartenance de la sortie qui correspond à la $i^{\text{ème}}$ règle ;

$\mu_i(x_1), \mu_i(x_2)$: Sont les facteurs d'appartenance des deux variables linguistiques aux deux ensembles flous de la $i^{\text{ème}}$ règle, pour deux valeurs données de x_1, x_2 ;

Z : est le nombre de règles.

II.3.4 Interface de défuzzification

La défuzzification consiste à déduire une valeur numérique précise de la sortie du régulateur (x_r) à partir de la conclusion résultante floue $\mu_{RES}(x_r)$ issue de l'opération d'inférence. Les méthodes couramment utilisées sont :

- La méthode du maximum ;
- La méthode de centre de gravité ;
- La méthode des surfaces ;
- La méthode des hauteurs.

II.3.4.1. Méthode du maximum

La sortie correspond à l'abscisse du maximum de la fonction d'appartenance résultante. Trois cas peuvent produire :

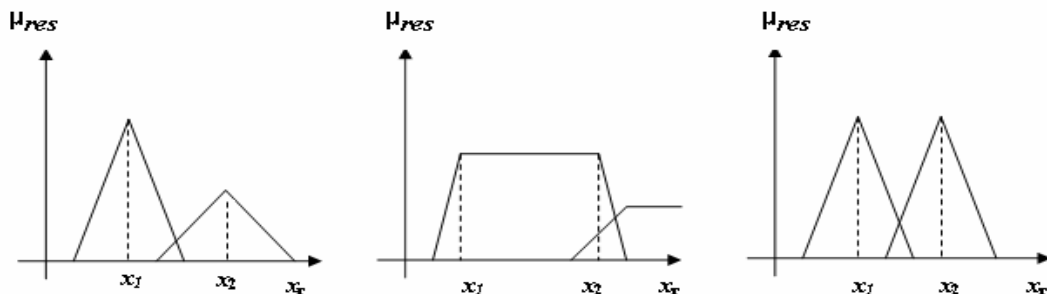


Figure II.8 : Défuzzification par valeur maximum.

- Dans le premier cas il n'y a pas de problèmes (car le maximum est x_1).

- Dans les deux autres cas, une ambiguïté apparaît. Il n'y a pas de règle générale sur la décision à prendre. Certains opérateurs préféreront prendre la plus petite sortie, d'autres la plus grande et d'autres une valeur entre x_1 et x_2 (uniquement pour le deuxième cas de la figure II.8).

II.3.4.2 La méthode du centre de gravité :

Cette méthode donne généralement de meilleurs résultats malgré l'exigence d'une grande puissance de calcul. Elle consiste à prendre comme décision à la sortie l'abscisse (x_{Gr}) du centre de gravité de la fonction d'appartenance résultante $\mu_{RES}(x_r)$ [11].

Cette abscisse est déterminée par la relation suivante :

$$x_{Gr} = \frac{\int_{-1}^1 x_r \mu_{RES}(x_r) dx_r}{\int_{-1}^1 \mu_{RES}(x_r) dx_r} \quad (II. 11)$$

Dans le cas de la méthode d'inférence somme-produit, on peut simplifier l'expression de $\mu_{RES}(x_r)$. En effet, selon la relation (I.6) on a :

$$\mu_{RES}(x_r) = \frac{1}{Z} \sum_{i=1}^Z \mu_{ci} \mu_{oi}(x_r) \quad (II. 12)$$

D'autre part, l'intégrale du dénominateur de (I.8) peut être simplifiée ainsi :

$$\int_{-1}^1 \mu_{RES}(x) dx_r = \frac{1}{Z} \sum_{i=1}^Z \mu_{ci} \int_{-1}^1 \mu_{oi}(x_r) dx_r = \frac{1}{Z} \sum_{i=1}^Z \mu_{ci} s_i \quad (II. 13)$$

Où s_i est la surface de la fonction d'appartenance du sous-ensemble flou de x_r correspondant à la $i^{ème}$ règle. Pour ce qui est de l'intégrale du numérateur y de (II.9), on peut la simplifier de la manière suivante :

$$\int_{-1}^1 x_r \mu_{RES}(x_r) dx_r = \frac{1}{Z} \sum_{i=1}^Z \mu_{ci} \int_{-1}^1 x_r \mu_{oi}(x_r) dx_r = \frac{1}{Z} \sum_{i=1}^Z \mu_{ci} x_{Gi} s_i \quad (II. 14)$$

Où x_{Gi} est l'abscisse du centre de gravité de la surface s_i .

On obtient finalement l'abscisse du centre de gravité de $\mu_{RES}(x_r)$ qui définit la commande ou l'action normalisée:

$$x_{Gr} = \frac{\sum_{i=1}^Z \mu_{ci} x_{Gi} s_i}{\sum_{i=1}^Z \mu_{ci} s_i} \quad (II. 15)$$

II.4. Avantages et inconvénients du réglage par logique floue

II.4.1. Avantages

Le réglage par logique floue présente les avantages suivants :

- La non nécessité modélisation (cependant il peut être utile d'un modèle convenable) ;
- La possibilité d'implémenter des connaissances (linguistiques) de l'opérateur du processus ;
- La maîtrise des systèmes à régler avec un comportement complexe (fortement non linéaire et difficile à modéliser) ;
- L'obtention fréquente de meilleures prestations dynamiques (régulateur non linéaire).
- L'emploi possible aussi pour des processus rapides (grâce à des processeurs dédiés).
- La disponibilité des systèmes de développement efficaces, soit pour microprocesseurs ou PC (solution logicielle). Soit pour circuits intégrés (processeurs dédiée, fuzzy processor, solution matérielle).

II.4.2. Les inconvénients

Les inconvénients liés à la commande floue sont résumés comme suit :

- Le manque de directives précises pour la conception d'un réglage (choix des grandeurs à mesurer, détermination de la fuzzification, des inférences et de défuzzification).
- L'approche artisanale et non systématique (implémentation des connaissances de l'opérateur souvent difficile).
- L'impossibilité de la démonstration de la stabilité du circuit de réglage en toute généralité (en l'absence d'un modèle valable).
- La possibilité d'apparition de cycles limites à cause du fonctionnement non linéaire ;
- La précision du réglage souvent peu élevée.
- La cohérence des inférences non garantie a priori (apparition de règles d'inférence contradictoires possible).

II.5 Conclusion

Dans ce chapitre, nous avons présenté les principaux axes concernant les fondements de la théorie des ensembles flous et de la logique floue. Nous avons ainsi exposé les différentes définitions des fonctions d'appartenances, règles d'inférences, méthodes d'inférences et la défuzzification.

De plus, nous avons introduit les différentes étapes de la commande floue, ceci nous permettra de bien définir les différentes composantes du contrôleur floue que nous allons utiliser dans notre application