

République Algérienne Démocratique et Populaire
Ministère de L'Enseignement Supérieur et de la A Recherche Scientifique

Université Mouloud Mammeri De Tizi-Ouzou



Faculté De Génie Electrique Et D'informatique
DEPARTEMENT D'AUTOMATIQUE

**Mémoire de Fin d'Etudes
de MASTER ACADEMIQUE**
Spécialité : **Automatique et informatique industrielle**

Présenté par
AMROUCHE Sylia
AZOUANI Djillali

Mémoire dirigé par **Mr TOUAT Mohand Achour**
Thème

**Commande neuro floue du canal
longitudinal d'un drone à voilure fixe.**

Mémoire soutenu publiquement le 09 juillet 2018 devant le jury composé de :

M Rabah MELLAH

Grade, Lieu d'exercice, Président

M Prénom NOM

Grade, Lieu d'exercice, Rapporteur

M Prénom NOM

Grade, Lieu d'exercice, Examineur

M Prénom NOM

Grade, Lieu d'exercice, Examineur

Remerciements

Nous tenons en premier lieu à remercier le bon Dieu pour le courage et la patience qu'il nous a donné an de mener ce projet à terme.

Pour la réalisation de ce mémoire, nous tenons à exprimer notre gratitude à Monsieur TOUAT Mohand Achour pour son encadrement riche en conseils et en encouragements, ainsi que sa disponibilité et pour toute l'aide qu'il nous a apportée tout au long de notre travail.

Nos plus vifs remerciements vont aussi aux membres du jury qui ont bien voulu nous honoré en acceptant d'examiner notre travail.

Nos vifs remerciements vont aussi à tous ceux qui ont contribué à la réalisation de ce travail et à tous ceux qui nous ont aidé à vaincre les nombreuses difficultés rencontrer toute au long de la durée de ce mémoire.

Nous exprimons, enfin, notre infinie gratitude à nos chers parents et à nos frères et sœurs en reconnaissance de leurs sacrifices, aides, soutiens et encouragements.

Djillali et Syla

Sommaire

Introduction générale:	1
I.1 Introduction	3
Chapitre I:	
I.2 La logique floue	4
I.3 Notions et Caractéristiques des sous-ensembles	4
I.3.1 Notions de sous-ensembles flous	4
I.3.2 Caractéristiques des sous-ensembles flous	4
I.4 Définitions sur les ensembles flous	5
I.4.1 Univers de discours	5
I.4.2 Variables linguistiques :.....	6
I.4.3 Fonctions d'appartenances :	6
I.5 Opérations sur les ensembles flous	8
I.6 Normes et conormes triangulaires	10
I.7 structure générale d'un contrôleur flou	11
I.7.1 la Fuzzification :.....	12
I.7.2 Base de connaissances floues :.....	13
I.7.2.1 règles d'inférence	13
I.7.2.2 Moteur d'inférences :	14
I.7.3 La Défuzzification :	15
I.7.3.1 La méthode de centre de gravité	15

I.7.3.2 La méthode de moyenne des maximums :	16
I.8 types de contrôleurs flous	16
I.8.1 Contrôleur Mamdani	16
I.8.2 Contrôleur TAKAGI SUGENO	17
I.9 Avantages et inconvénients de la logique floue	18
I.9.1 Les Avantages :	18
I.9.2 les inconvénients :	18
I.10 Conclusion	19

Chapitre II:

II.1. Introduction :	20
II.2. Neurone biologique :	20
II.3. Neurone artificiel (formel) :	20
II.4. Les perceptrons :	22
II.5. Les différentes architectures des réseaux de neurones :	23
II.5.1. Les réseaux non bouclés monocouche :	23
II.5.2. Les réseaux non bouclés multi couche :	24
II.5.3. Les réseaux de neurones bouclés :	24
II.5.4. Les réseaux de neurones à connexions complexes :	25
II.6. Apprentissage des réseaux de neurones :	25
II.6.1. Processus d'apprentissage :	25
II.6.1.1. Apprentissage supervisé :	26
II.6.1.2. Apprentissage non supervisé :	26

II.6.1.3. Apprentissage hybride :	27
II.7. Quelques méthodes d'apprentissage :	27
II.7.1. La règle delta :	27
II.7.2. Apprentissage par retro-propagation du gradient de l'erreur :.....	29
II.8. Conclusion :	31

Chapitre III:

III.1 Introduction	32
III.2 Définitions :	32
III.3 les méthodes de combinaison neuro flous :	33
III.3.1 Réseau neuro-flou concurrent :	33
III.3.2 Réseau neuro-flou coopératifs :.....	34
III.3.4 Les systèmes neuro- flous hybrides :.....	34
III.4 Types des systèmes neuro flous.....	34
III.5 Le modèle ANFIS	35
III.5.1 Architecture de l'ANFIS :	35
III.5.2 Apprentissage du réseau ANFIS :	38
III.5.3 Les avantages de l'ANFIS.....	40
III.6 Conclusion.....	40

Chapitre IV:

IV.1. Introduction.....	41
IV.2 Description de la dynamique du vol :.....	41

IV.3 Géométrie générale d'un avion :	42
IV.3.1. Fuselage :	42
IV.3.2. L'aile	43
IV.3.3. Les gouverne :	43
IV.4. Les repères de référence :	44
IV.4.1. Trièdre terrestre $R_0(OX_0Y_0Z_0)$:	44
IV.4.2. Trièdre aérodynamique $R_s(OX_sY_sZ_s)$:	44
IV.4.3. Trièdre avion $R_a(OX_aY_aZ_a)$:	44
IV.5. Passage entre repères :	45
IV.6 Les forces appliquées sur un avion	45
IV.7 Équations de mouvement	47
IV.8 La théorie des petites perturbations	49
IV.9 Linéarisation de la dynamique du vol	50
IV.10 Equations linéarisées du mouvement longitudinal	53
IV.11 Conclusion	54

Chapitre V:

V.1 Introduction :	55
V.2 Schéma bloc du système sur Simulink :	55
V.3 Simulation et interprétation des résultats :	56
V.4 Conclusion :	60
Conclusion générale :	61

Liste des figures

Figure I.1 : représentation des caractéristiques de la logique floue.....	15
Figure I.2 : Exemple de représentation graphique des termes linguistiques	15
Figure I.3 : Représentation des variables binaires.....	16
Figure I.4 : Représentation des variables floues.....	17
Figure I.5 : fonction d'appartenance singleton.....	18
Figure I.6 : Intersection des sous-ensembles flous « petite» et «grande » de la variable linguistique (vitesse).....	19
Figure I.8 : Complémentation du sous-ensemble flou «petite».....	20
Figure I.9 : Schéma général d'un contrôleur flou.	22
Figure I.11 Défuzzification par la méthode moyenne maximum.....	26
Figure II.1 : Neurone biologique.	30
Figure II.2 : Neurone artificiel.....	31
Figure II.3 : fonctions d'activation les plus utilisées.	32
Figure II.4 : Perceptron à seuil.....	32
Figure II.5 : Apprentissage de la fonction AND et XOR.....	33
Figure II.6 : réseau mono couche.....	33
Figure II.7 : réseau non bouclé multi couche.....	34
Figure II.8 : Réseau de neurone bouclé	35
Figure II.10 : Apprentissage non supervisé.	37
Figure II.11 Réseau de neurone a deux couches	39
Figure III.1 : Structure générale d'un réseau neuro-flou.....	42
Figure III.2 : Principe du système neuro-flou.....	43

Figure III.3 Système neuro-flou concurrent.....	43
Figure III.4 : Système neuro-flou Coopératif	44
Figure III .5 Architecture de FALCON	45
Figure III.6 : L'Architecture de l'ANFIS.	46
Figure III.7 : Exemple ANFIS à 2 entrées avec 9 règles.	48
Figure IV.1 Représentation des six degrés de liberté d'un avion.....	51
Figure IV.2 : les composants d'un avion	52
Figure IV.3 : Géométrie de l'aile	53
Figure IV.4 : Représentation des trois principaux axes	53
Figure IV.5 : Forces subies par l'avion	56
Figure V.1: Model SIMULINK du système	65
Figure V.2: . Model SIMULINK du correcteur neuro flou ANFIS	66
Figure V.3: vitesse de translation en fonction du temps.	67
Figure V.4: l'altitude en fonction du temps	68
Figure V.5: l'angle d'attaque alpha en fonction du temps	68
Figure V.6: l'angle de tangage θ en fonction du temps.	69
Figure V.7: La variation de l'angle de tangage en fonction du temps.....	70

Liste des tableaux

Tableau I.1 : règles d'inférence	23
Tableau III.1 : les différentes couches d'un système ANFIS	47
Tableau IV.1. Variable mise en jeu lors de la modélisation	51

Introduction générale

Introduction générale :

L'avion est le moyen de transport aérien le plus rapide, plus sûr et le plus confortable. Il change de direction très souvent et obéit à des perturbations atmosphériques qui influencent sur la qualité du vol, pour cela on a développé des commandes pour avoir une bonne qualité de vol.

La description du vol montre une grande complexité due aux non linéarités ainsi que la difficulté de modéliser parfaitement la dynamique du vol, le redressement et la stabilisation de ce dernier deviennent ardues car les connaissances du système se révèlent imprécises et imparfaites.

Par ailleurs les problèmes de représentation et d'utilisation des connaissances sont au centre d'une discipline scientifique relativement nouvelle : l'intelligence artificielle qui a mis l'accent de façon exclusive sur le traitement symbolique des connaissances, plus récemment on a assisté à un retour du numérique dans les problèmes de l'intelligence artificielle avec les réseaux de neurone et la logique floue, alors que les réseaux de neurone proposent une approche implicite de la représentation des connaissances. La logique floue permet de faire un lien entre le numérique et la symbolique à partir de simples algorithmes de traduction.

L'approche traditionnelle pour la conception floue est basée sur des connaissances acquises par des experts formulés sous forme de règles. Cependant il se peut que les experts ne puissent pas transcrire leurs connaissances sous forme de contrôleur flou, c'est à partir de ces constatations qu'on a développé l'approche connexionniste qui combine les capacités d'apprentissage des réseaux de neurones et les systèmes flous pour former des systèmes neuro-flou ce qui permet d'exploiter les caractéristiques de chacune pour accomplir une tâche performante et efficace pour la commande des systèmes complexes.

L'objectif de notre travail porte sur l'analyse et la modélisation de la commande du vol d'un avion à base des stratégies de la commande NEURO-FLOU qui vise à minimiser l'énergie mise en jeu et qui permet de maintenir l'avion sur un angle de tangage constant pour cela nous avons organisé notre travail comme suit :

- Le premier est consacré à la présentation de la logique floue.
- Le deuxième chapitre est réservé aux définitions des réseaux de neurones.
- Le troisième chapitre est dédié aux systèmes neuro-flous.
- Le quatrième chapitre traite la dynamique du vol.

Introduction générale

- Le chapitre cinq donne les résultats de simulation ainsi que les commentaires et les conclusions sur ces mêmes résultats.

Enfin, nous terminons ce travail par une conclusion générale.

I.1 Introduction

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

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

La majorité des systèmes industriels complexes sont difficiles à contrôler.

Cette difficulté provient de :

- leur non-linéarité,
- la variation de leurs paramètres,
- la qualité des variables mesurables.

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

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

Nous présentons dans ce chapitre les concepts de la logique floue et nous décrivons les aspects méthodologiques nécessaires à la compréhension de cette technique en donnant quelques concepts de la logique floue et les différents types de la commande en utilisant cette technique.

I.2 La logique floue

La logique floue se base sur la théorie des ensembles vise à représenter les connaissances incertaines et imprécises et fournit un moyen approximatif mais efficace pour décrire le comportement des systèmes qui sont trop complexes ou mal définis. La logique floue utilise des instructions progressives plutôt que la logique (booléenne) qui est strictement vrai ou faux.

La conception d'un système basée sur la logique floue commence par le choix des variables linguistiques (entrées et sorties du contrôleur), le pas suivant consiste à définir l'univers du discours pour chaque variable linguistique, c'est-à-dire un ensemble de valeurs que la variable peut prendre.

I.3 Notions et Caractéristiques des sous-ensembles

I.3.1 Notions de sous-ensembles flous [3]

Dans la théorie classique des ensembles, un sous ensemble A de X est défini par une fonction caractéristique $\mu_A(x)$ qui prend la valeur 0 pour les éléments de X n'appartenant pas à A et la valeur 1 pour ceux qui appartiennent à A :

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

Un sous ensemble flou A de X est défini par une fonction d'appartenance qui associe à chaque élément x de X, le degré $\mu_A(x)$, compris entre 0 et 1 avec lequel x appartient à A :

$$\mu_A(x) \in [0 \ 1]$$

I.3.2 Caractéristiques des sous-ensembles flous [5]

- **Support** : c'est-à-dire l'ensemble des éléments de X qui appartiennent, au moins un peu, à A. Il est noté $\text{supp}(A)$ et c'est la partie de X sur laquelle la fonction d'appartenance de A n'est pas nulle : $\text{supp}(A) = \{x \in X / \mu_A(x) \neq 0\}$
- **Hauteur** : notée $h(A)$, c'est-à-dire le plus fort degré avec lequel un élément de X appartient à A. C'est la plus grande valeur prise par sa fonction d'appartenance:
$$h(A) = \sup_{x \in X} \mu_A(x)$$

- **Noyau** : c'est l'ensemble de tous les éléments d'appartenant de façon absolue (avec un degré 1) à A et noté $\text{noy}(A)$: $\text{noy}(A) = \{x \in X / \mu_A(x) = 1\}$
- **Cardinalité** : évaluant le degré global avec lequel les éléments de X appartiennent à A. Elle est définie par : $|A| = \sum_{x \in X} \mu_A(x)$.

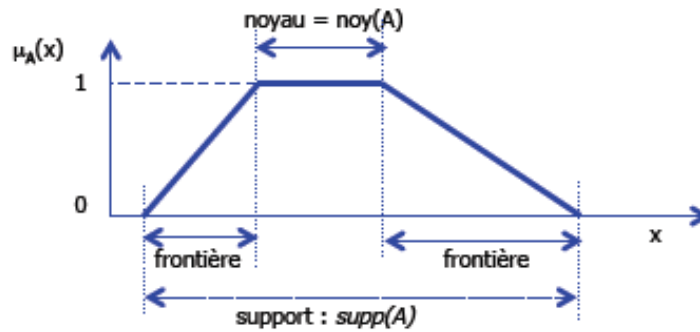


Figure I.1 : représentation des caractéristiques de la logique floue

I.4 Définitions sur les ensembles flous

I.4.1 Univers de discours [1]

Un des premiers pas dans la conception d'une application floue est de définir l'ensemble de référence ou l'univers de discours pour chaque variable linguistique.

L'univers de discours représente l'ensemble de référence ou le domaine de variation de la variable linguistique, ou domaine de fonctionnement du processus dans le cas de réglage, comme illustré sur la figure ci-dessous dont plusieurs sous-ensembles sont décrit par des fonctions d'appartenance μ_A triangulaires et trapézoïdales.

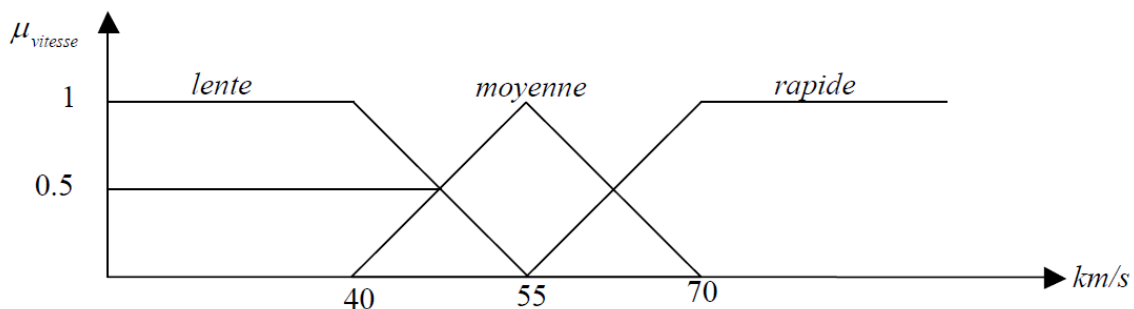


Figure I.2: Exemple de représentation graphique des termes linguistiques

I.4.2 Variables linguistiques :

Les variables linguistiques d'un mot est un ensemble de termes qui évoquent le même concept mais à degrés différents. Il peut être fini ou non.

Une variable linguistique représente un état du système à régler.

Sa valeur est définie avec des termes linguistique qui peuvent être des mots ou des phrases d'un Langage naturel ou artificiel : « quelque, beaucoup, souvent », « négatif, zéro, positif ».

I.4.3 Fonctions d'appartenances :

L'ensemble des variables linguistiques que peut prendre la variable floue sont représentées par une fonction d'appartenance et chaque élément de cet ensemble a une valeur d'appartenance qui est le degré de compatibilité de cet élément avec le concept qui est représenté par l'ensemble flou. Cette valeur varie entre 0 et 1 contrairement à la logique classique qui a deux valeurs seulement 0 ou 1.

Afin de mieux saisir la différence qui existe entre les deux logiques et de mettre en évidence le principe fondamental de la logique floue, présentons un exemple simple.

On se propose de classifier des vitesses en fonction de leurs valeurs en définissant trois catégories comme suit :

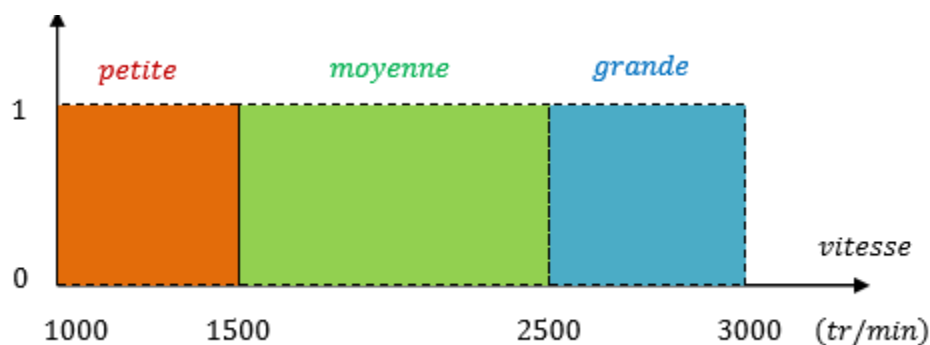


Figure I.3 : Représentation des variables binaires

On constate que cette façon utilisée pour séparer les intervalles des vitesses est très éloigné de ce que fait l'être humain lorsqu'il analyse ce genre de situation. En effet, l'homme ne fait pas une distinction franche entre « petite » et « moyenne » par exemple. Il utilise des expressions telles que « plutôt petite » pour une vitesse légèrement inférieure à 1500 (tr/min), et « plutôt moyenne » pour une vitesse juste supérieure à cette valeur. Donc la logique classique présente bien l'avantage de la simplicité, mais elle est relativement éloignée de la logique utilisée par l'être humain.

Chapitre I : La logique floue

Représentons le même problème à l'aide de la logique floue. Les variables ne sont plus de nature binaire mais peuvent prendre une infinité de valeurs possibles entre 0 et 1. La figure suivante représente la classification considérée selon le principe de la logique floue.

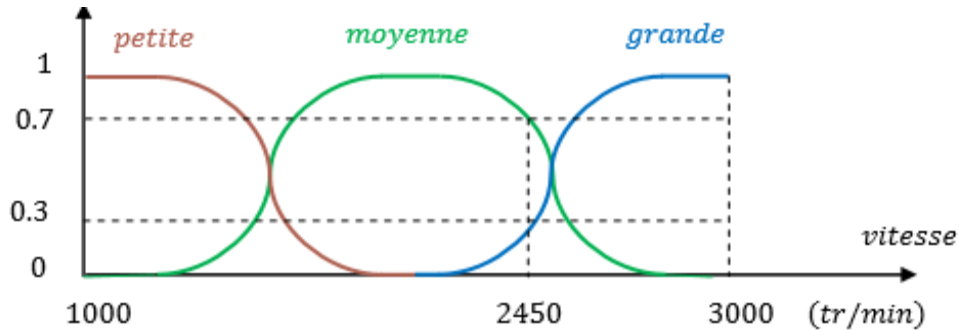
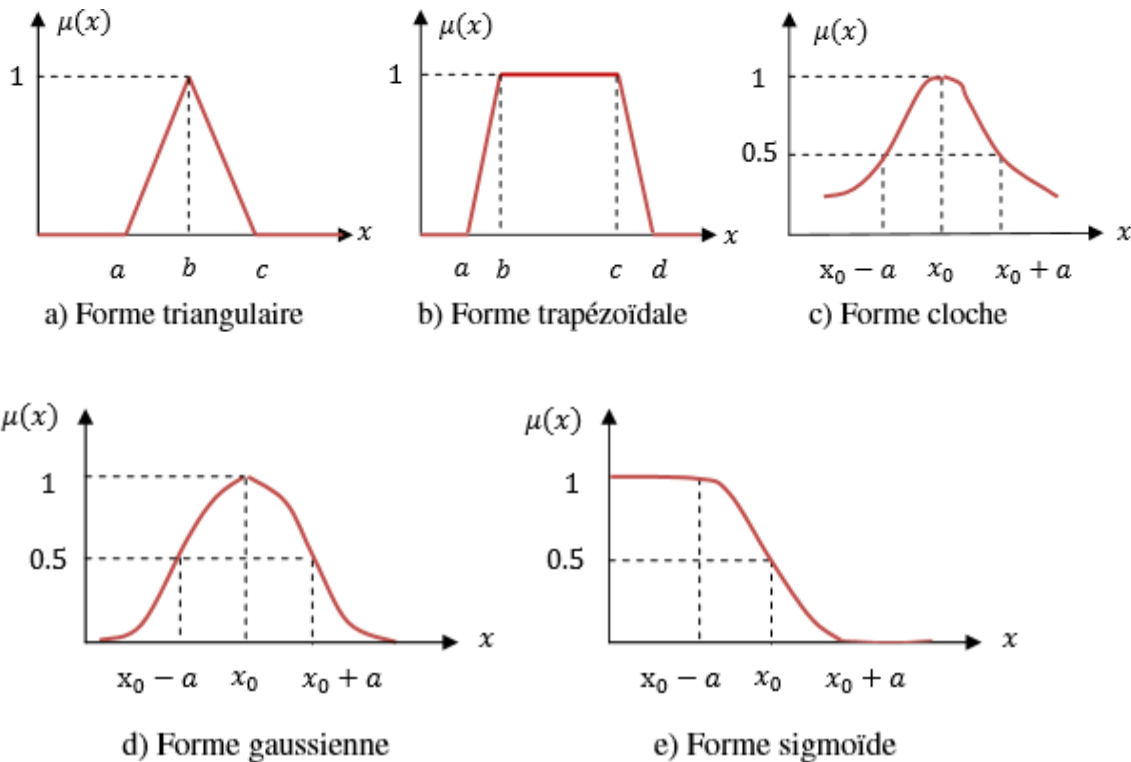


Figure I.4 : Représentation des variables floues

La fonction d'appartenance peut être représentée par plusieurs formes : Triangulaire, trapézoïdale, sigmoïdale, cloche et gaussienne. On peut définir d'autres formes de fonctions d'appartenance, mais dans le réglage par logique floue, les formes déjà citées sont largement suffisantes pour délimiter les ensembles flous.



Pour certaines situations, les fonctions d'appartenance sont égales à l'unité pour une seule valeur de la variable et égale à 0 pour les autres comme le montre la figure. Elles prennent

alors le nom de « fonction d'appartenance singleton ». Elle correspond dans le domaine flou à une valeur particulière de cette variable.

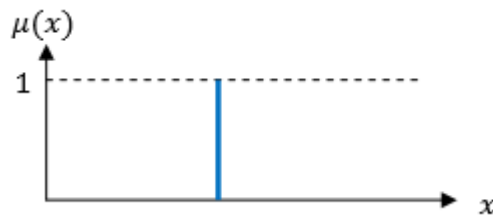


Figure I.5 : fonction d'appartenance singleton

I.5 Opérations sur les ensembles flous [5]

Ces opérateurs permettent d'écrire des combinaisons logiques entre notions floues, c'est-à-dire de faire des calculs sur des degrés de vérité. Comme pour la logique classique. Il existe de nombreuses variantes dans ces opérateurs. Cependant, les plus répandus sont ceux dits « de Zadeh ». Leur utilisation sera reprise dans l'exemple didactique d'utilisation d'une base de règles floues.

Dans ce qui suit, le degré de vérité d'une proposition A sera noté $\mu(A)$:

- **Egalité**

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

$$\forall x \in X \mu_A(x) = \mu_B(x)$$

- **Inclusion**

A est dit inclus dans B, propriété que l'on note $A \subset B$, si tout élément x de X qui appartient, même de façon modérée, à A appartient aussi à B avec un degré au moins aussi grand. Leurs fonctions d'appartenances sont telles que:

$$\forall x \in X \mu_A(x) \leq \mu_B(x)$$

- **Intersection (ET)**

L'intersection de deux sous-ensembles flous A et B de X est le sous ensemble flou de X qui contient tous les éléments x de X appartenant à la fois à A et B. l'intersection de deux sous-ensembles flous A et B ($A \cap B$) de X est le sous ensemble flou C tel que

$$\forall x \in X ; \mu_C(x) = \min(\mu_A(x), \mu_B(x))$$

Ou

$$\mu_C(x) = \mu_A(x) \cdot \mu_B(x)$$

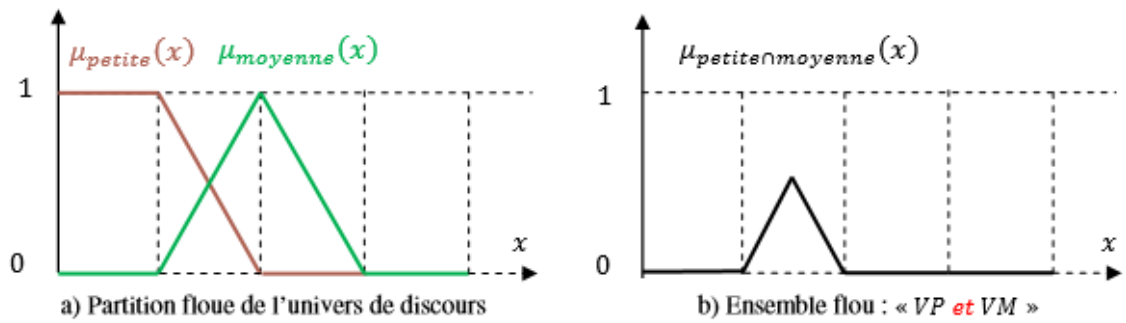


Figure I.6: Intersection des sous-ensembles flous « petite » et « grande » de la variable linguistique (vitesse)

- **Union (ou)**

L'union de deux sous ensemble flous A et B de X est le sous ensemble flou de X qui contient tous les éléments appartenant ou bien a A ou bien a B. l'union de deux sous-ensembles flou A et B ($A \cup B$) de X est le sous ensemble flou D de X tel que

$$\forall x \in X; \mu_D(x) = \max(\mu_A(x), \mu_B(x))$$

Ou

$$\mu_D(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \cdot \mu_B(x)$$

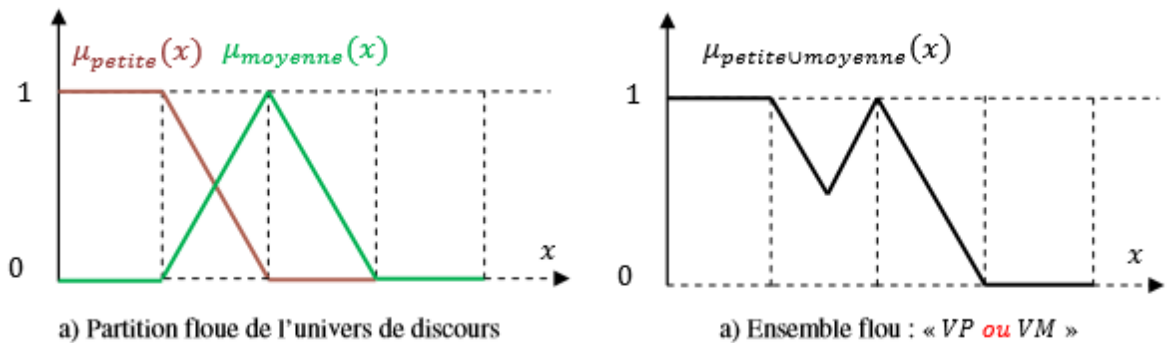


Figure I.7 : Union des sous-ensembles flous « petite » et « grande » de la variable linguistique (vitesse)

- **Propriétés de l'union et de l'intersection**

Comme dans la théorie des ensembles classique, les définitions que nous venons de donner conduisent aux propriétés suivantes :

- i. Associativité de \cap et de \cup ,
- ii. Commutativité de \cap et de \cup ,
- iii. $A \cup \emptyset = A$, $A \cup X = X$,

- iv. $A \cap X = A, A \cap \emptyset = \emptyset,$
- v. $A \cap (B' \cup B'') = (A \cap B') \cup (A \cap B''),$
- vi. $A \cup (B' \cap B'') = (A \cup B') \cap (A \cup B'')$

Et la cardinalité de sous-ensembles flous vérifie: $|A| + |B| = |A \cap B| + |A \cup B|.$

- **Complément d'un sous ensemble flou (NON)**

Soit un sous-ensemble flou A de X , son complément est le sous ensemble contenant tous le x n'appartenant pas à A . Le complément A^C d'un sous ensemble flou A de X est défini comme le sous ensemble flou de X de fonction d'appartenance :

$\forall x \in X; \mu_{A^C}(x) = 1 - \mu_A(x)$ on peut énoncer les propriétés suivantes $A^C \cap A = \emptyset$ et $A^C \cup A = X$

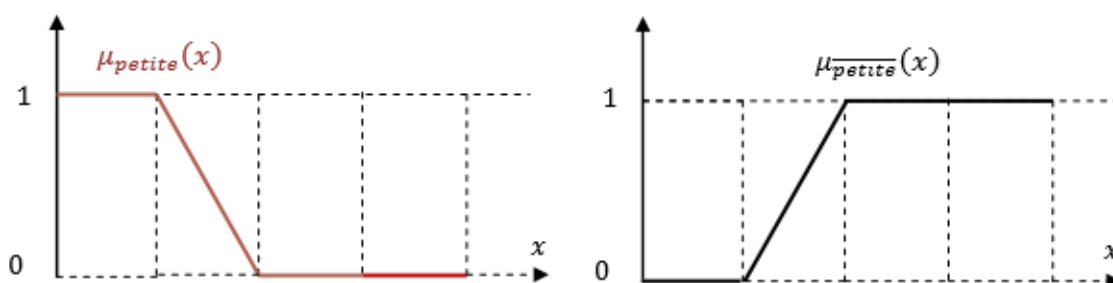


Figure I.8: Complémentation du sous-ensemble flou «petite».

- **Produit cartésien**

Le produit cartésien de r sous - ensembles flous A_1, \dots, A_r , définis respectivement sur X_1, \dots, X_r , est le sous ensemble A de X de fonction d'appartenance:

$$\forall x = (x_1, \dots, x_r), \mu_A(x) = (\mu_{A_1}(x), \dots, \mu_{A_r}(x))$$

I.6 Normes et conormes triangulaires [5]

Les opérations d'intersection, d'union et de complémentation usuelles pour les sous-ensembles flous peuvent être effectuées à l'aide d'opérateurs autres que le minimum, le maximum, la complémentation a1.

Des familles de telles opérations ont été introduites, à l'origine dans le domaine des espaces métriques probabilisés. La logique floue, la commande floue de processus, par exemple, font appel à de tels opérateurs. Ces familles, sont définies de la façon suivante.

Chapitre I : La logique floue

Une *norme triangulaire (t-normes)* est une fonction $T : [0, 1] \times [0, 1]$ qui vérifie pour tous x , y et z de $[0, 1]$:

- i. $T(x, y) = T(y, x)$ (commutativité),
- ii. $T(x, T(y, z)) = T(T(x, y), z)$ (associativité),
- iii. $T(x, y) \leq T(z, t)$ si $x \leq z$ et $y \leq t$ (monotonie),
- iv. $T(x, 1) = x$ (élément neutre 1).

Cas particulier : l'opérateur $T = \min$ est une norme triangulaire.

Une *conorme triangulaire (t-conorme)* est une fonction $\perp : [0, 1] \times [0, 1]$ qui vérifie pour tous x , y et z de $[0, 1]$:

- i. $\perp(x, y) = \perp(y, x)$ (commutativité),
- ii. $\perp(x, \perp(y, z)) = \perp(\perp(x, y), z)$ (associativité),
- iii. $\perp(x, y) \leq \perp(z, t)$ si $x \leq z$ et $y \leq t$ (monotonie),
- iv. $\perp(x, 0) = x$ (élément neutre 0).

Cas particulier l'opérateur $\perp = \max$ est une conorme triangulaire.

Toute t-normes et toute t-conormes vérifient: $T(0, 0) = 0$, $T(1, 1) = 1$, $\perp(0, 0) = 0$, $\perp(1, 1) = 1$.

I.7 structure générale d'un contrôleur flou

Un contrôleur flou n'est autre qu'un système flou conçu pour commander un processus. Sa structure générale est traduite par le schéma donné par la *figure I.9*, La représentation des connaissances est basée sur les ensembles flous dont les règles fournissent directement les conclusions. L'interface avec le flou assure la conversion des variables linguistiques du système en variables floues compréhensibles par le régulateur (Fuzzification). Le contrôleur flou, grâce à sa base de règle génère alors la conclusion qui sera rendu compréhensible pour le système par l'interface non floue (Défuzzification).

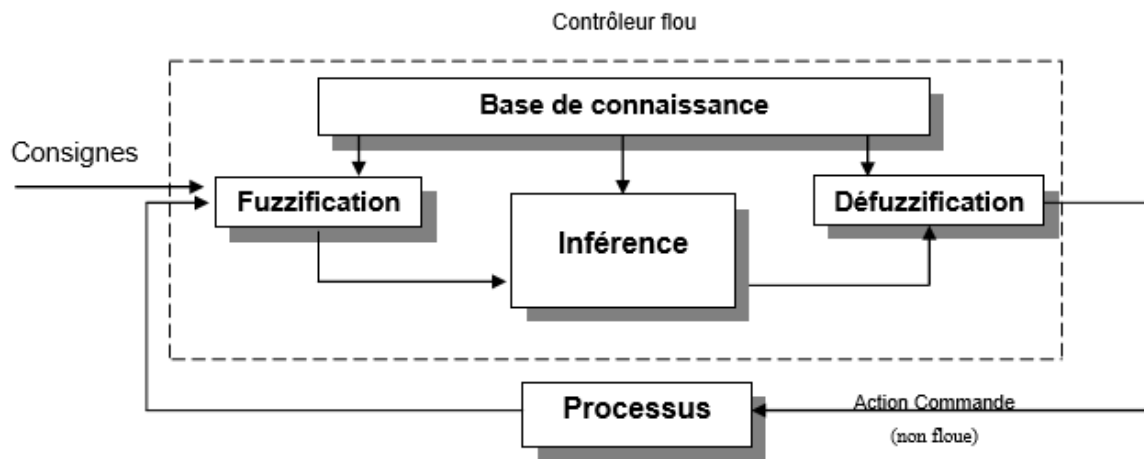


Figure I.9 : Schéma général d'un contrôleur flou.

I.7.1 la Fuzzification :

La Fuzzification est une opération qui transforme une entrée numérique observée en une valeur linguistique, on effectue cette transformation par attribution de degrés d'appartenance à chaque valeur d'entrée. Les variables linguistiques sont définies par leurs valeurs linguistiques. En générale, les fonctions d'appartenance qui représentent les valeurs linguistiques sont définies en forme triangulaire, trapézoïdale ou en forme gaussienne.

Il n'y a pas de règle précise pour la définition de fonction d'appartenance. On peut introduire pour une variable x plusieurs valeurs linguistiques.

Les fonctions d'appartenance peuvent être symétriques, régulièrement distribués ou avoir une distribution non uniforme. Il est important d'éviter les chevauchements insuffisants de deux ensembles voisins. Ceci provoque des zones mortes.

On distingue deux types de fuzzification respectivement, appelées fuzzification numérique et fuzzification symbolique. Les désignations standard des ensembles flous est :

- **NG** Negative Grand
- **NM** Negative Moyen
- **NP** Négative Petit
- **EZ** Environ Zéro
- **PP** Positif Petit
- **PM** Positif Moyen
- **PG** Positif Grand

Chapitre I : La logique floue

Un nombre de valeurs linguistiques supérieur à sept n'apporte en général aucune amélioration de comportement dynamique du réglage flou par contre un tel choix compliquerait la formulation des règles d'inférences.

Les fonctions d'appartenance peuvent être symétriques, régulièrement distribuées ou avoir une distribution non uniforme.

I.7.2 Base de connaissances floues :

Elle est constituée d'une base de données et une base de règles. La base de données contient les fonctions d'appartenance des ensembles flous, et la base de règles est une collection de règles floues IF-THEN qui définissent la relation entre une antécédente représentant la description de l'état de système et la conséquente exprimant l'action de l'opérateur qui contrôle le système.

I.7.2.1 règles d'inférence [2] :

On appelle règles d'inférence l'ensemble des différentes règles reliant les variables floues d'entrée d'un système aux variables floues de sorties de ce système à l'aide des différents opérateurs. Ces règles se présentent sous la forme suivante :

Si condition 1 ET/OU condition 2 (ET/OU...) alors action sur les sorties OU

Si condition 3 ET/OU condition 4 (ET/OU...) alors action sur les sorties OU

⋮

Si condition n ET/OU condition n+1 (ET/OU...) alors action sur les sorties.

Ou par une matrice d'inférence sous formes d'un tableau donné ci-dessous :

X		x		
		N	EZ	P
Dx/dt	N	N	N	EZ
	EZ	N	EZ	P
	P	EZ	P	P

Tableau I.1 : règles d'inférence

On distingue deux types de règles:

1. Règles à conclusion symbolique (contrôleur de type Mamdani) : La sortie est une valeur linguistique. Exemple : *Si l'erreur est « Négatif » et la variation de l'erreur est « Positif » Alors la commande est « zéro ».*
2. Règles à conclusion algébrique (contrôleur de Sugeno) : la sortie est une valeur numérique (singleton) ou une équation mathématique bien précise (non floue). Exemple: *Si l'erreur est « Négatif » et la variation de l'erreur est « Positif » Alors la commande est -0.3.* Lorsque la sortie est une valeur numérique on parle de règles de Takagi-Sugeno « d'ordre zéro », sinon, de règles à conclusion polynomiale.

I.7.2.2 Moteur d'inférences :

Il transforme à l'aide des techniques de raisonnement flou, la partie floue issue de la fuzzification en une nouvelle partie floue, afin d'évaluer le degré de vérité d'une règle i à partir des fonctions d'appartenances liées à la prémisse et la conclusion. L'agrégation des n règles s'effectue en combinant les différentes règles floues à l'aide de l'opérateur OU.

Il existe plusieurs méthodes pour réaliser ces opérations à savoir :

- **Méthode d'inférence Max-Min**

Cette méthode réalise l'opérateur "ET" par la fonction "Min", la conclusion

"ALORS" de chaque règle par la fonction "Min" et la liaison entre toutes les règles (opérateur « OU ») par la fonction Max. La dénomination de cette méthode, dite Max-Min ou

"implication de Mamdani", est due à la façon de réaliser les opérateurs ALORS et OU de l'inférence [4].

- **Méthode d'inférence Somme-Produit**

L'opérateur "ET" est réalisé par le produit, de même que la conclusion "ALORS".

Cependant, l'opérateur "OU" est réalisé par la valeur moyenne des degrés d'appartenance intervenant dans l'inférence. D'autres méthodes ont été élaborées ayant chacune une variante spécifique. Néanmoins, la méthode Max-Min est de loin la plus utilisée à cause de sa simplicité [4].

I.7.3 La Défuzzification :

La défuzzification consiste à transformer l'ensemble flou résultant de l'agrégation des règles en une grandeur de commande précise. Plusieurs méthodes de défuzzification.[6] peuvent être utilisées:

- La méthode de la hauteur,
- Le premier des maximas,
- Le dernier des maximas,
- La moyenne des maximas,
- Le centre de gravité,
- Le centre des aires,
- Le centre de la plus grande surface,
- Le centre des maximas.

Les méthodes de défuzzification les plus utilisées en commande floue sont le centre de gravité, le centre des aires et le centre des maximas.

I.7.3.1 La méthode de centre de gravité

La méthode de centre de gravité est une des méthodes les plus mentionnées dans la littérature. L'abscisse du centre de gravité peut être déterminée en utilisant la forme générale :

$$z = \frac{\int_{z_0}^{z_1} \mu_z(z) * z dz}{\int_{z_0}^{z_1} \mu_z(z) dz} \quad (\text{I.1})$$

Lorsque la fonction $\mu(z)$ est discrétisée, le centre de gravité est donné par :

$$z = \frac{\sum_{i=1}^n \mu_i(z) * z_i}{\sum_{i=1}^n \mu_i(z)} \quad (\mathbf{I.2})$$

Ou n est le nombre de règles, z_i est la valeur de sortie pour le niveau i et μ_i sa valeur d'appartenance.[6]

I.7.3.2 La méthode de moyenne des maximums :

Cette méthode génère une commande précise en calculant la moyenne des valeurs pour lesquelles l'appartenance est maximale

$$z = \sum_{i=1}^l \frac{r_i}{l} \quad (\mathbf{I.3})$$

Ou l est le nombre de valeurs quantifiées de r pour lesquelles l'appartenance est maximale.

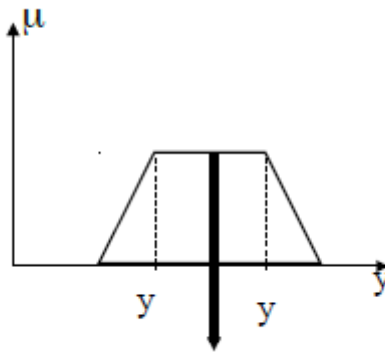


Figure I.11 Défuzzification par la méthode moyenne maximum

I.8 types de contrôleurs flous

En se basant sur les différentes règles de commande floue et de leurs méthodes de génération, les approches de commande en utilisant la logique floue peuvent être classées dans les catégories suivantes :

I.8.1 Contrôleur Mamdani [7]

Dans un système flou de Mamdani (Conventionnel) les règles sont de type :

$$\mathbf{R}_{(i,j,k)} : \mathbf{Si} \ x \text{ est } A_i \text{ et } y \text{ est } B_j \text{ alors } z \text{ est } C_k$$

Ou A_i , B_j et C_k représentent des termes linguistiques auxquels est attribuée une signification floue que l'on suppose normalisée.

Chapitre I : La logique floue

Pour des entrées numériques précises, x_0 et y_0 , un sous-ensemble flou de sortie, noté F , est généré selon la relation :

$$\mu_f(z) = \max_{(i,j,k) \in I} \min(\mu_{A_i}(x_0), \mu_{B_j}(y_0), \mu_{C_k}(z))$$

μ est appelée fonction d'appartenance.

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

$$z = \frac{\int z \mu_f(z) dz}{\int \mu_f(z) dz} \quad (\mathbf{I.4})$$

Dans ce type de systèmes flous, la prémisse et la conclusion sont floues. Après l'inférence, le résultat est un ensemble flou caractérisé par sa fonction d'appartenance. Afin d'obtenir la valeur réelle de la commande à appliquer il faut passer par une étape de « défuzzification ».

Les règles de la commande floue « Si-Alors » sont obtenues à partir de l'expertise d'un opérateur. Quoique la performance d'une telle technique de commande soit généralement satisfaisante en pratique le non garanti de la stabilité du système de commande en boucle fermée est souvent critiquée.

Plusieurs approches ont été proposées pour étudier le problème de stabilité. L'idée principale de ces approches est de considérer le contrôleur flou comme un contrôleur non linéaire et d'utiliser la théorie des systèmes non linéaire pour l'analyse de la stabilité.

I.8.2 Contrôleur TAKAGI SUGENO

Les contrôleurs flous de Takagi Sugeno sont, comme ceux de Mamdani, construits à partir d'une base de règles "Si ... Alors ...". Les prémisses sont toujours exprimées linguistiquement et donc similaires à celles utilisées dans un contrôleur flou de Mamdani alors que les conclusions sont de nature numérique.

Un système flou de type Takagi-Sugeno (T-S) utilise des règles écrites de la manière suivante:

$$\mathbf{R}_{(i,j,k)} : \text{Si } x \text{ est } A_i \text{ et } y \text{ est } B_j \text{ alors } z \text{ est } f_k(x)$$

Où f_k représente une fonction réelle quelconque.

Les fonctions de sortie f_k , peuvent être en principe des fonctions arbitraires des entrées, mais d'une manière générale elles sont choisies telles qu'elles soient une combinaison linéaire des entrées.

La particularité d'un modèle T-S est que la logique floue est seulement utilisée dans la partie prémisses des règles. La partie conclusion est décrite par des valeurs numériques.

Pour les valeurs d'entrée précises x_0 et y_0 , la sortie z est évaluée selon le mécanisme suivant :

$$z = \frac{\sum_{(i,j,k) \in I} w_{i,j} \cdot f_k(x_0, y_0)}{\sum_{(i,j,k) \in I} w_{i,j}} \text{ avec } W_{i,j} = \mu_{A_i}(x_0) \cdot \mu_{B_j}(y_0)$$

Ce type de modèle est aussi très intéressant pour la représentation de systèmes non linéaires.[8]

I.9 Avantages et inconvénients de la logique floue

I.9.1 Les Avantages :

- La non nécessité d'une modélisation mathématique. Cependant, il peut être utile de disposer d'un modèle convenable.
- La maîtrise du système à régler avec un comportement complexe.
- La possibilité d'implanter des connaissances (linguistiques) de l'opérateur de processus.
- La simplicité de définition et de conception.
- L'obtention fréquente de meilleures prestations dynamiques (régulateur non linéaire).

I.9.2 les inconvénients :

- Le manque de directives précises pour la conception d'un réglage (choix des grandeurs à mesurer, détermination de la fuzzification, des inférences et de la défuzzification).
- L'approche artisanale et non systématique (implantation des connaissances des opérateurs 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 cohérence des inférences non garantie a priori (possibilité d'apparition de règles d'inférence contradictoire).

I.10 Conclusion

Dans ce chapitre, nous avons présenté la théorie sur laquelle repose la logique floue et nous avons vu au cours de ces définitions que le concepteur d'un système flou doit faire un nombre de choix important et suivre quelques étapes dans l'ordre. Ces choix essentiellement sont basés sur les conseils de l'expert ou sur l'analyse statistique des données passées.

La théorie de la logique floue nous permet de modéliser les systèmes qui ont des modèles mathématiques compliqués, ils exploitent des règles floues tirées d'une expertise humaine pour modéliser le comportement dynamique du système.

II.1. Introduction :

Les réseaux de neurones artificiels (RNA) sont des réseaux fortement connectés de processeurs élémentaires fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu'il reçoit. Ils infèrent des propriétés émergentes permettant de solutionner les problèmes jadis qualifiés de complexe.

II.2. Neurone biologique :

Un neurone biologique est une cellule capable de transmettre des informations à d'autres neurones à travers ces différentes connexions (synapses). Il existe plusieurs types de neurones (pyramides, paniers,.....) avec des fonctionnements différents (sensoriels, moteur, inter-neurones). Ces neurones reçoivent des impulsions électriques par les dendrites et envoient l'information par les axones, le contact entre ces deux derniers se fait par des synapses.

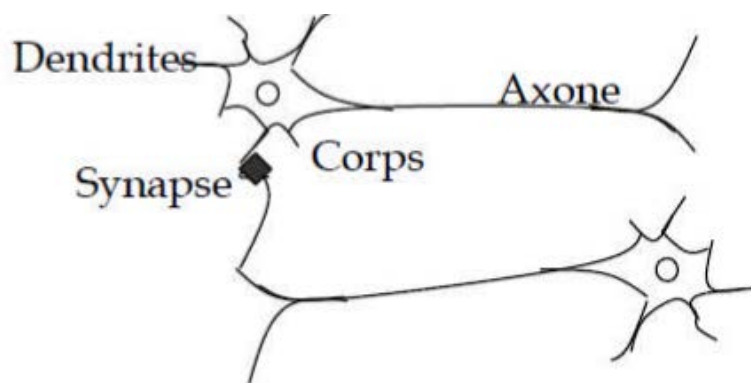


Figure II.1 : Neurone biologique.

II.3. Neurone artificiel (formel) :

Une analogie directe faite par Colluche et Pitt avec le neurone biologique a donné naissance au neurone formel qui est un processeur qui applique des opérations simples à ses entrées et que l'on peut relier à d'autres pour former un réseau qui peut réaliser des relations entrées sorties [9].

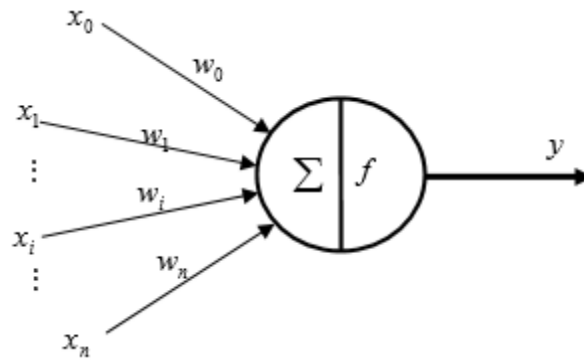


Figure II.2 : Neurone artificiel.

$$Z = \sum_{i=0}^n X_i W_i \text{ et } y = f(z) \text{ II.1}$$

Les entrées du neurone sont désignées par les X_i ($i=1 \dots n$), et les paramètres W_i sont les poids synaptiques, f est la fonction d'activation qui est généralement croissante et bornée. Les réseaux de neurones font partie des réseaux adaptatifs non linéaires, dans la mesure où la structure des neurones est modifiée lors du processus d'apprentissage. Un RNA est composé de petites unités de traitement interconnectées. Les informations sont transmises à travers ces unités le long des interconnexions. Une connexion à l'entrée a deux valeurs qui lui sont associées, une valeur d'entrée et un poids. La sortie du neurone est une fonction de la somme des entrées multipliées par leurs poids respectifs. En fait, les RNA sont entraînés avec le jeu de données jusqu'à ce qu'ils apprennent le comportement décrit par les données à l'entrée.

Les fonctions d'activation les plus utilisées sont les suivantes :

- | | |
|----------------------------|----------------------------|
| a) tout ou rien ; | e) saturation ; |
| b) fonction signe ; | f) sigmoïde ; |
| c) plus ou moins à seuil ; | g) fonction arc tangente ; |
| d) fonction affine ; | h) fonction gaussienne. |

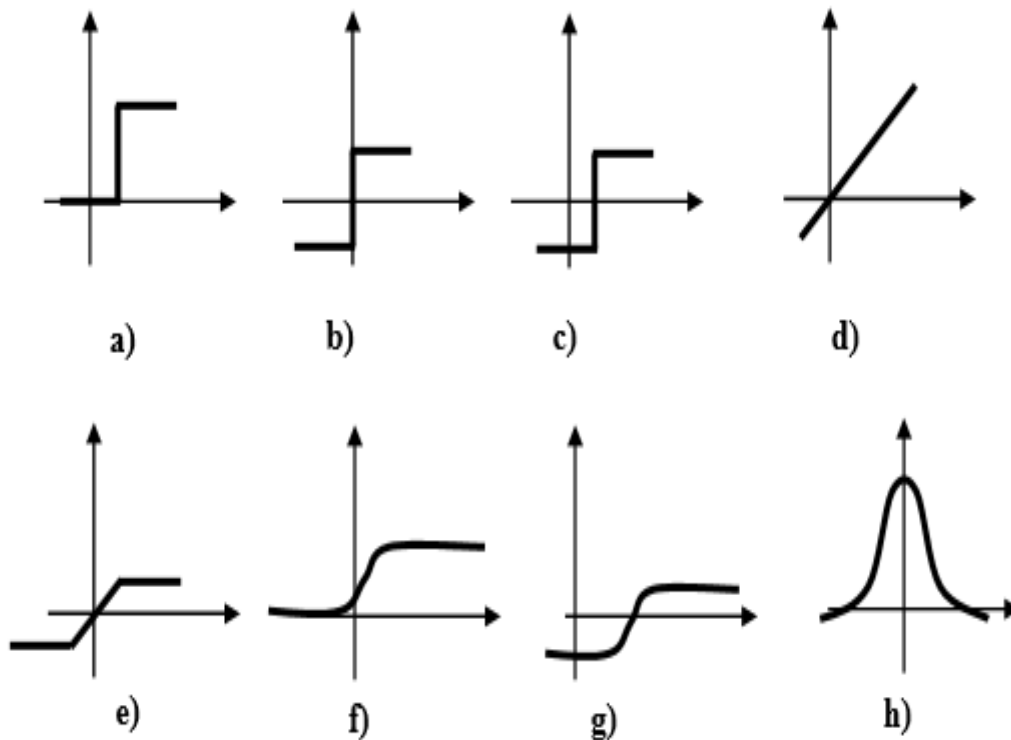


Figure II.3 : fonctions d'activation les plus utilisées.

II.4. Les perceptrons :

Un perceptron est un RNA à propagation direct (feedforward en anglais) qui a été introduit pour la première fois en 1958[10]. Le perceptron utilise le processus d'apprentissage supervisé pour résoudre des opérations logiques de base comme AND ou OR. Néanmoins, il ne peut pas résoudre des opérations logiques plus complexes, telles que la fonction XOR. Il est souvent utilisé à des fins de classification [11].

Un perceptron linéaire à seuil prend en entrée $x_1 \dots x_n$, n valeurs et calcule une sortie o , les coefficients synaptiques $w_1 \dots w_n$ et le seuil (ou le biais) comme illustré dans la figure suivante :

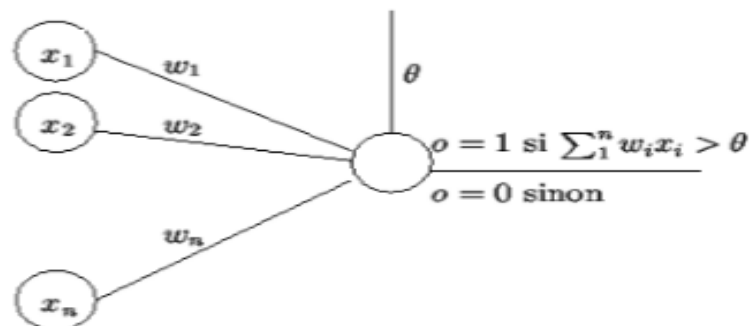


Figure II.4 : Perceptron à seuil

Exemple :

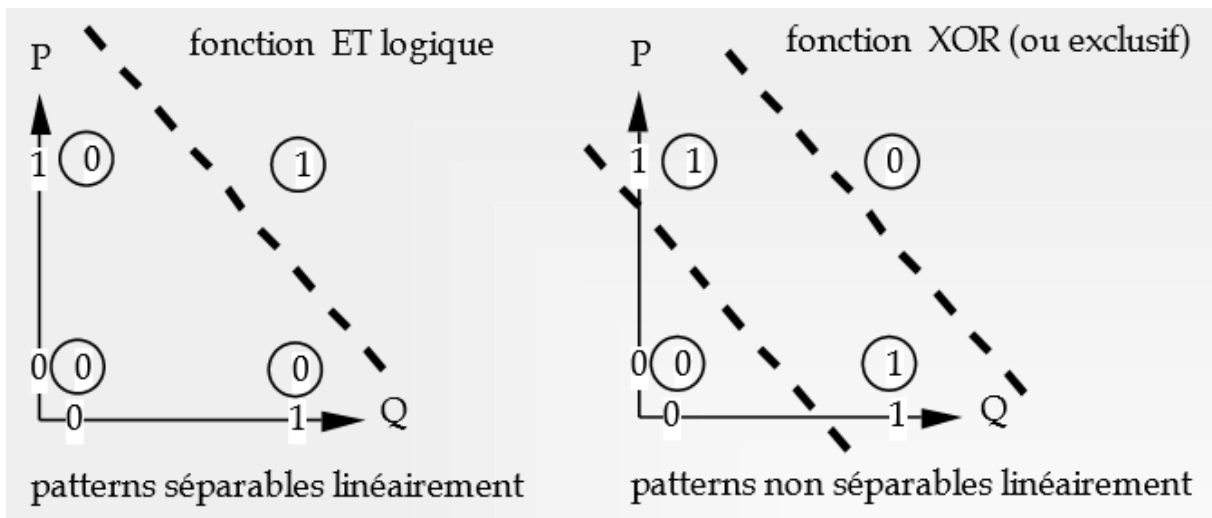


Figure II.5 : Apprentissage de la fonction AND et XOR.

II.5. Les différentes architectures des réseaux de neurones :

II.5.1. Les réseaux non bouclés monocouche :

C'est un réseau simple unidirectionnel, il se compose d'une couche d'entrée et d'une couche de sortie. La couche d'entrée reçoit les stimuli à traiter par l'intermédiaire des nœuds sources. Cette couche se projette en une couche de sortie composée de neurones (nœuds de calcul) transmettant les résultats du traitement au milieu extérieur. La figure suivante présente un réseau proactif Mono Couche a 4 nœuds d'entrée et 3 nœuds de sortie. La désignation Monocouche est attribuée à la couche de sortie.

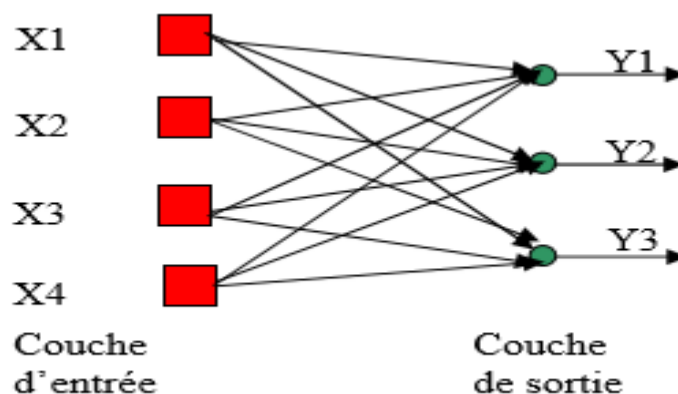


Figure II.6 : réseau mono couche.

II.5.2. Les réseaux non bouclés multi couche :

Les perceptrons multi couche (MLP) unidirectionnel sont les RNA les plus utilisés, ils sont constitués de plusieurs couches, dont les nœuds de calcul correspondants s'appellent neurones cachés. Les couches cachées s'interposent entre l'entrée du réseau et sa sortie. Leurs rôle est d'effectuer un prétraitement des signaux d'entrée, reçus par la couche d'entrée en provenance du milieu extérieur, et de transmettre les résultats correspondants à la couche de sortie ou seront déterminées les réponses finales du réseau. Grâce à sa structure étendue, le MLP peut résoudre des problèmes plus complexes et toutes les fonctions logiques y compris la fonction XOR. Car en ajoutant une ou plusieurs couches cachées, le réseau est capable d'extraire plus de propriétés statistiques que celles extraites d'un réseau similaire ayant moins de couches cachées. Ceci est utile pour réaliser des fonctions plus complexes que de simples séparations linéaires.

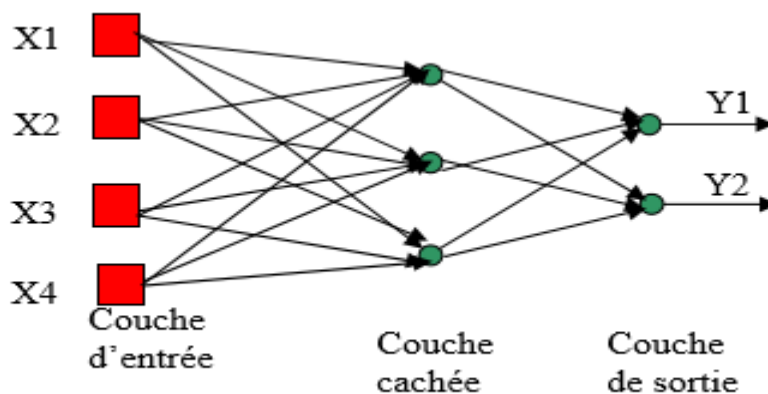


Figure II.7: réseau non bouclé multi couche.

II.5.3. Les réseaux de neurones bouclés :

Contrairement aux réseaux de neurones non bouclés dont les connexions sont acycliques, les réseaux de neurones bouclés peuvent avoir une topologie de connexion quelconque comprenant notamment des boucles qui ramène aux entrées la valeur d'une ou plusieurs sorties. Ces réseaux sont assez puissants, car leur fonctionnement est séquentiel et adopte un comportement dynamique.

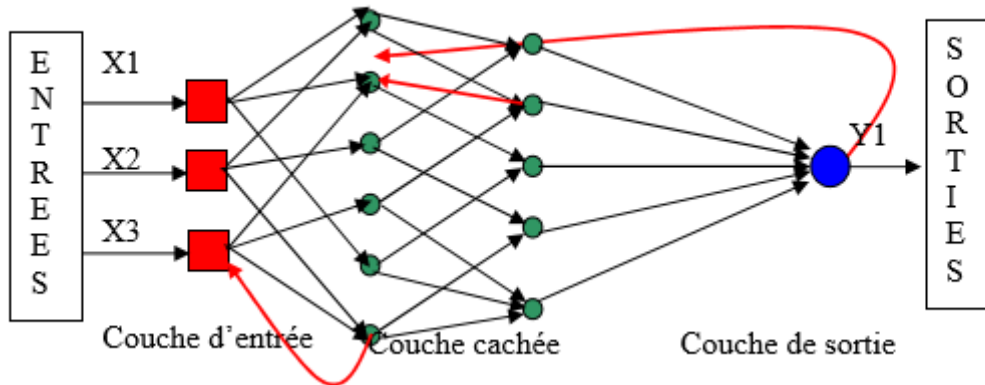


Figure II.8 : Réseau de neurone bouclé

II.5.4. Les réseaux de neurones à connexions complexes :

Dans la catégorie des réseaux à connexions complexes se trouvent tous les réseaux qui ne trouvent pas leur place dans les deux catégories précédemment citées. Nous pouvons mentionner les SOM (Self Organizing Map), LVQ (Learning Vector Quantization), ART (Adaptative Resonance Théorie), RCE (Restrict-Ted Coulomb Energy), les réseaux probabilistes, etc [12].

II.6. Apprentissage des réseaux de neurones :

Une fois l'architecture d'un réseau de neurones choisie, il est nécessaire d'effectuer un apprentissage pour déterminer les valeurs des poids permettant à la sortie du réseau de neurones d'être aussi proche que possible de l'objectif fixé. Dans le cas d'un problème de régression, il s'agit d'approcher une fonction continue, dans le cas d'un problème de classification supervisée, il s'agit de déterminer une surface de séparation.

II.6.1. Processus d'apprentissage :

Dans les années 60 on ne savait pas réaliser un apprentissage sur un réseau à rebouclage ou sur un réseau à plusieurs couches. Cela mit en sommeil les recherches sur les réseaux neuronaux plusieurs années. Ce n'est qu'au début des années 80 que les travaux ont conduit à des méthodes d'apprentissages autorisant l'utilisation des réseaux de neurones complexes.

L'apprentissage est la propriété principale des RNA, c'est une procédure qui consiste à estimer les paramètres du réseau afin que celui-ci remplisse au mieux la tâche qui lui est affectée. Au cours d'apprentissage le réseau corrige sa structure en augmentant ou en

diminuant ou en annulant les valeurs de connexion entre ces neurones. On distingue trois types d'apprentissages :

II.6.1.1. Apprentissage supervisé :

Ce mode dispose d'un comportement de référence, vers lequel il tente de converger la sortie du réseau. Durant une première phase qui est la phase d'apprentissage, on présente au réseau les valeurs d'entrée et on calcule sa sortie actuelle correspondante, on suite on ajuste les poids de façon à réduire l'écart entre la sortie désirée et celle du réseau en utilisant l'erreur de sortie. Après, vient la phase de test on présente au réseau de nouvelles valeurs et l'on observe ses réponses. Les performances du réseau sont évaluées par le pourcentage de l'erreur. Donc l'apprentissage supervisé n'est possible que lorsque la conduite du système est connue ou fixé par un expert.

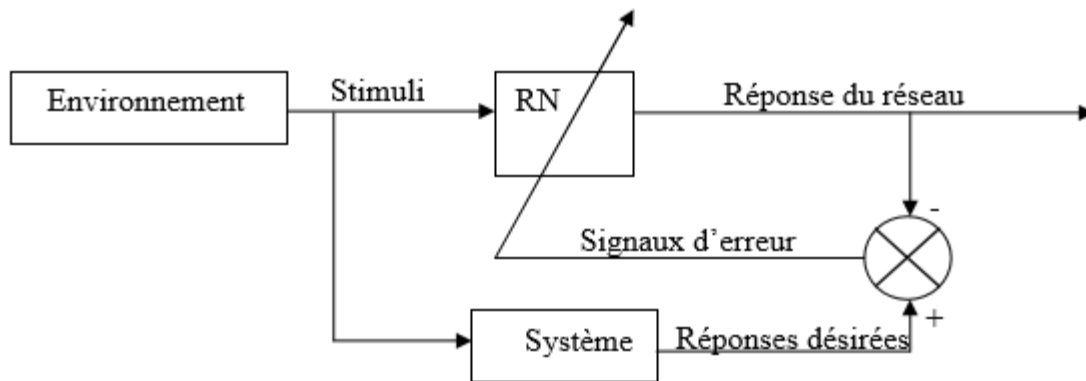


Figure II.9 : Apprentissage supervisé.

II.6.1.2. Apprentissage non supervisé :

Contrairement aux modes supervisés, seule une base d'entrées est fournie au réseau. Celui-ci doit déterminer lui-même ses sorties en fonction des similarités détectées entre les différentes entrées, c'est-à-dire en fonction d'une règle d'auto-organisation. Il explore la structure des données, ou encore la corrélation entre les modèles d'entrée et les organise à partir de cette corrélation en des catégories.

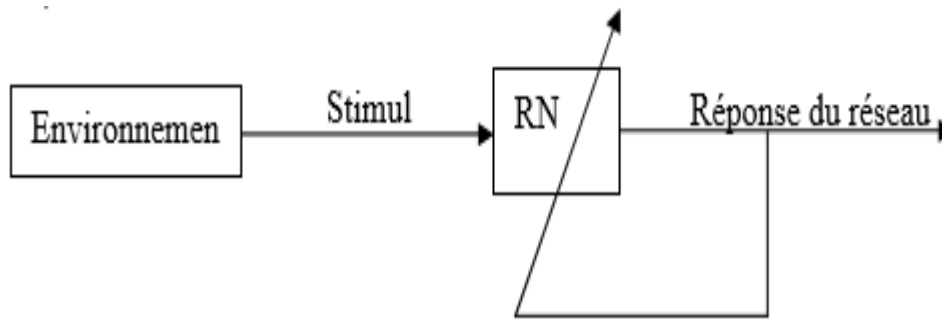


Figure II.10 : Apprentissage non supervisé.

II.6.1.3. Apprentissage hybride :

Il combine l'apprentissage supervisé et non supervisé. Dans de nombreux problèmes on ne dispose pas de l'information nécessaire à la construction d'une base d'apprentissage complète. On ne dispose souvent que d'une information qualitative permettant l'évaluation de la réponse calculée, sans pour autant connaître la réponse la plus adaptée.

II.7. Quelques méthodes d'apprentissage :

La plupart des méthodes d'apprentissage sont des algorithmes d'optimisation, elles cherchent à minimiser une fonction de coût qui constitue une mesure de l'écart entre les réponses réelles du réseau et ces réponses désirées. Cette optimisation se fait d'une manière itérative, en modifiant les poids synaptique, et il existe plusieurs algorithmes pour faire l'apprentissage.

II.7.1. La règle delta :

L'erreur E est la fonction de la variable w_{ij} $i=1,2,\dots,m$; $j=1,2,\dots,n$. On rappelle que le gradient ∇E de E au point w est le vecteur des dérivées partielles $\frac{\partial E}{\partial w_{ij}}$, et comme la dérivée d'une fonction a une seule variable, le gradient pointe toujours vers la direction du battement de E la descente la plus raide et E en w est $-\nabla E$, donc pour minimiser E on doit se déplacer proportionnellement au négatif de ∇E pour la mise à jour de chaque poids w_{ij} comme suit :

$$w_{jk} \rightarrow w_{jk} + \Delta w_{jk} \text{II.2}$$

D'où
$$\Delta w_{ik} = -\eta \frac{\partial E}{\partial w_{ij}} \text{II.3}$$

Chapitre II : Les réseaux de neurones

Et η est une constante appelée pas d'apprentissage

On a:

$$\frac{\partial E}{\partial w_{ij}} = \sum_{q=1}^n \frac{\partial E^q}{\partial w_{ij}} \quad \text{II.4}$$

Et

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial}{\partial w_{ij}} (\sum_{i=1}^m (o_i^q - y_i^q)^2) \quad \text{II.5}$$

$$= (o_i^q - y_i^q) \frac{\partial}{\partial w_{ij}} f_j (\sum_{i=0}^n w_{ji} x_i^q) \quad \text{II.6}$$

Puisque

$$\frac{\partial}{\partial w_{ij}} (o_i^q - y_i^q) = 0 \quad \text{pour } i \neq j \quad \text{II.7}$$

Rien que

$$o_j^q = f_j (\sum_{i=0}^n w_{ji} x_i^q) \quad \text{II.8}$$

Donc

$$\frac{\partial E^q}{\partial w_{ij}} = x_k^q (o_j^q - y_j^q) f_j' (\sum_{i=0}^n w_{ji} x_i^q) \quad \text{II.9}$$

$$= \delta_j^q x_k^q \quad \text{II.10}$$

D'où

$$\delta_j^q = (o_j^q - y_j^q) f_j' (\sum_{i=0}^n w_{ji} x_i^q) \quad \text{II.11}$$

Pour la sortie du j éme neurone

$$\text{D'où } \Delta \omega_{ik} = \sum_{q=1}^n \Delta^q w_{jk} \quad \text{II.12}$$

$$= -\eta \sum_{q=1}^n \frac{\partial E^q}{\partial w_{ik}} \quad \text{II.13}$$

$$= \sum_{q=1}^n -\eta \delta_j^q x_k^q \quad \text{II.14}$$

Chapitre II : Les réseaux de neurones

La règle delta vise à trouver le vecteur des poids w^* , tel que le gradient de E a w^* est 0, pour quelques réseaux de neurones Monocouche w^* est la seule minimum de E (w).

II.7.2. Apprentissage par retro-propagation du gradient de l'erreur :

Considérons les réseaux de neurones multicouches pour l'approximation de la relation générale, nous avons donc besoin d'algorithmes d'apprentissage pour ces réseaux plus complexes. L'algorithme de propagation populaire suivant est une généralisation de la règle delta appliquée aux réseaux de neurones monocouche.

La mise a jour des poids se fait comme suit :

$$\Delta w_{ik} = -\eta \delta_i^q x_k^q \quad \text{II.15}$$

Pour cela, on doit calculer la fonction :

$$\delta_i^q = (O_i^q - y_i^q) f'(\sum_{j=0}^n w_{ij} x_j^q) \quad \text{II.16}$$

Ceci est possible puisque on a la valeur de y_i^q connue comme sortie du réseau de neurone.

Considérons réseau de neurone a deux couche $n - m - p$ comme le montre la figure suivante :

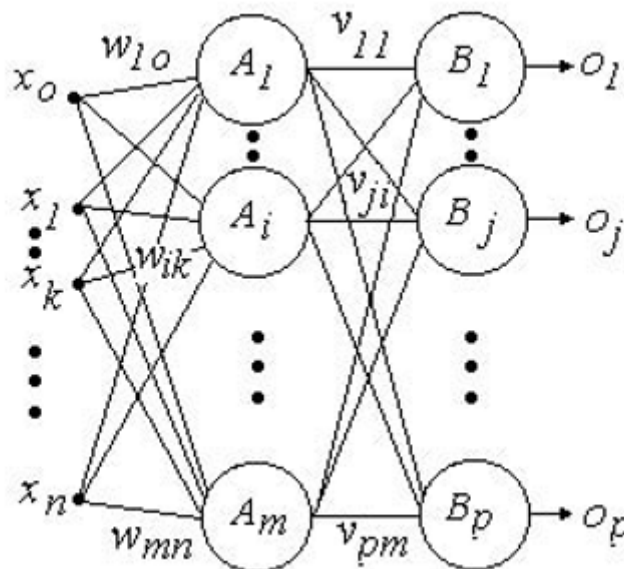


Figure II.11 Réseau de neurone a deux couches

Notons que y^q est un modèle dans la couche de sortie et non pas celle cachée, par conséquent lorsque nous regardons aux erreurs $(O_i^q - y_i^q)^2$ dans la couche de sortie nous ne pouvons pas détecter la couche responsable de cette erreur.

Chapitre II : Les réseaux de neurones

Il s'avère que pour mettre à jour ces poids il suffit de calculer $\frac{\partial E}{\partial O_i}$ la dérivée partielle de l'erreur globale E par rapport à la sortie $O_i, O_i = 1, \dots, p$. Pour les fonctions d'activation différentiables, la stratégie de descente du gradient peut être utilisée pour trouver les configurations de poids w^* ce qui minimise l'erreur E.

Nous écrirons les équations de mise à jour des poids dans un réseau de neurone à deux couches, la généralisation à plus de deux couches est juste une question de notation.

Soit v_{ji} le poids reliant le neurone caché i au neurone de sortie j, et w_{ik} le poids reliant le nœud d'entrée k au neurone caché i.

Premièrement la règle de mise à jour pour le v_{ji} est la même qu'en règle delta mais en considérant la couche cachée en tant que couche d'entrée. Pour le neurone de sortie j et le neurone caché i, le poids v_{ji} s'écrit comme suit :

$$\Delta v_{ji} = \sum_{j=1}^N \Delta^q v_{ji} \quad \text{II.17}$$

$$= -\eta \sum_{j=1}^N \frac{\partial E^q}{\partial v_{ji}} \quad \text{II.18}$$

$$= \sum_{j=1}^N (-\eta \delta_j^q z_i^q) \quad \text{II.19}$$

Où z_i^q est l'entrée du neurone caché i.

$$z_i^q = f_i(\sum_{k=0}^n w_{ik} x_k^q) \quad \text{II.20}$$

Et

$$\delta_j^q = (O_j^q - y_j^q) f'_i(\sum_{k=1}^m v_{jk} z_k^q) \quad \text{II.21}$$

Pour le neurone caché i, et l'entrée du nœud k, le poids w_{ik} est donné comme suit :

$$\Delta^q w_{ik} = -\eta \frac{\partial E^q}{\partial w_{ik}} \quad \text{II.22}$$

Et

$$\frac{\partial E^q}{\partial w_{ik}} = \frac{\partial E^q}{\partial O_i^q} \cdot \frac{\partial O_i^q}{\partial w_{ik}} \quad \text{II.23}$$

Où

O_i^q est la sortie du neurone caché i

$$O_i^q = f_i(\sum_{l=0}^n w_{il}x_l^q) = z_i^q \text{ II.24}$$

On a

$$\frac{\partial O_i^q}{\partial w_{ik}} = f_i'(\sum_{l=0}^n w_{il}x_l^q)x_k^q \text{ II.25}$$

$$\text{On pose } \delta_i^q = \frac{\partial E^q}{\partial O_i^q} \text{ II.26}$$

$$= \sum_{j=1}^p \frac{\partial E^q}{\partial O_j^q} \cdot \frac{\partial O_j^q}{\partial O_i^q} \text{ II.27}$$

O_j : la sortie de la couche cachée.

Le $\frac{\partial E^q}{\partial O_i^q}$ est calculé précédemment (règle de delta)

$$O_i^q = f_j'(\sum_{l=1}^m v_{il}z_l^q)v_{ji} \text{ II.28}$$

Ainsi les δ_i^q , pour le neurone caché i , sont calculés à partir de valeurs déjà connues de δ_j^q pour tout j dans la couche de sortie.

De ce fait la règle delta généralisée est appelée algorithme de retro propagation inverse, d'abord alimentent les motifs d'entrée x^q vers l'avant pour atteindre la couche de sortie, puis calculent les δ_j^q pour tous les neurones de sortie j . Ensuite propager ces δ_j^q vers l'arrière dans la couche précédente afin de calculer les δ_i^q pour tous les neurones i de cette couche.

II.8. Conclusion :

Le développement qu'ont connu les réseaux de neurones artificiels a suscité un grand enthousiasme mais aussi des critiques. Beaucoup d'études comparatives comme celle effectuée dans ce mémoire ont fournies une vue optimiste pour les réseaux de neurones artificiels, tandis que d'autres offrent une vue pessimiste. Pour de nombreuses tâches et problèmes, tel que la reconnaissance de formes, il n'y a pas une seule approche qui pourrait dominer les autres. Le choix de la technique doit être en fonction de l'application en question, c'est pour ça qu'on doit comprendre et connaître les capacités, suppositions, la possibilité d'appliquer plusieurs approches dans différentes disciplines.

III.1 Introduction

Les systèmes Neuro-Flous permettent de combiner les avantages de deux techniques complémentaires. Les systèmes flous fournissent une bonne représentation des connaissances.

L'intégration de réseaux de neurones améliore leurs performances grâce à la capacité d'apprentissage de réseaux de neurones. Inversement, l'injection des règles floues dans les réseaux de neurones, souvent critiqués pour leur manque de lisibilité, clarifie la signification des paramètres du réseau et facilite leur initialisation, ce qui représente un gain de temps de calcul considérable pour leur identification.

De nombreux types de systèmes Neuro-flous ont été définis et développés ces dernières années, loin d'être uniformisés, et parfois équivoques et confus.

Afin de clarifier les définitions, nous proposons dans ce chapitre une présentation brève de quelques types de systèmes Neuro-flous et une présentation plus détaillée de l'ANFIS.

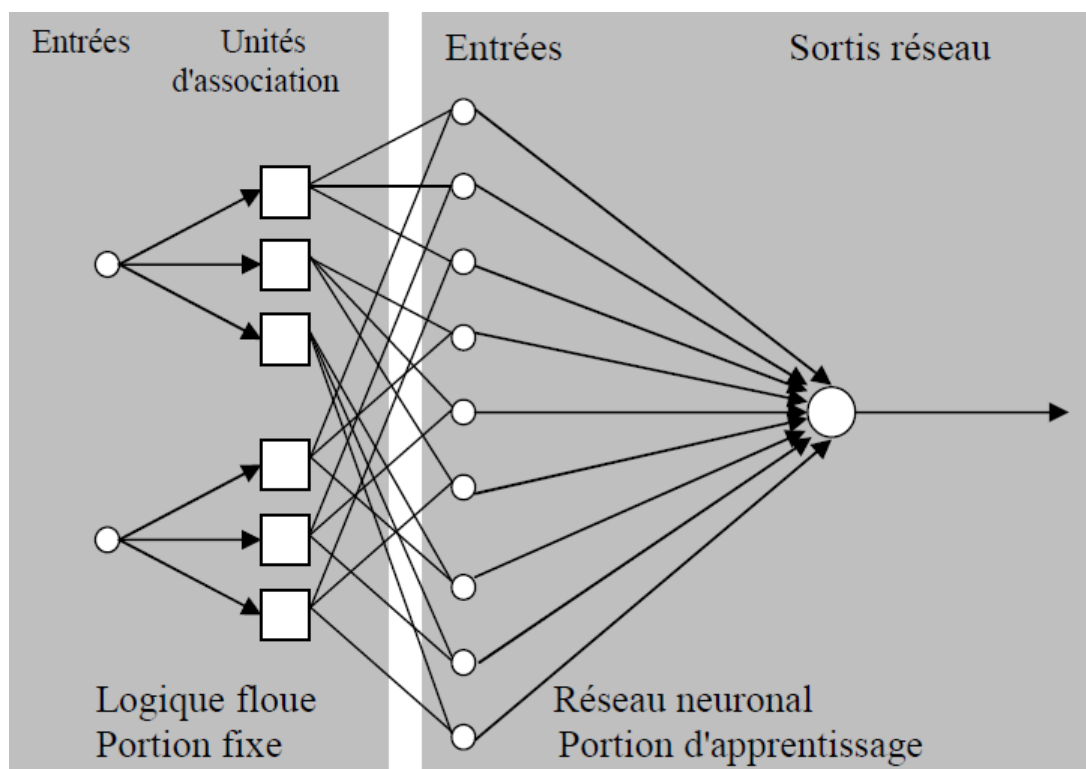


Figure III.1 : Structure générale d'un réseau neuro-flou.

III.2 Définitions :

On définit un système neuro-flou comme étant un réseau neuronal multicouche avec des paramètres flous, ou comme un système flou mis en application sous une forme distribuée.[13]

Chapitre III : Systèmes neuro-flous

Un système neuro-flou ne devrait par contre pas être vu comme un système expert mais ils peuvent être utilisés comme des approximateurs universels.

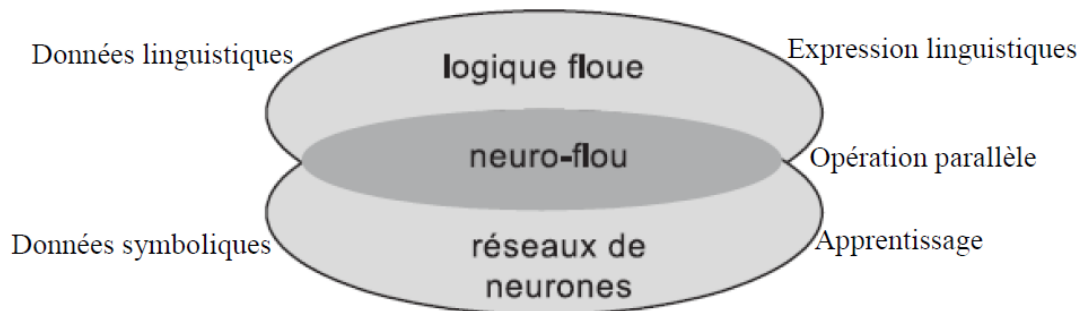


Figure III.2 : Principe du système neuro-flou.

III.3 les méthodes de combinaison neuro flous :

Il y a plusieurs types pour combiner les réseaux de neurone et les systèmes flous. Ces types peuvent être classés en fonctionnelle et structurelle, suivant leur architecture et la configuration recherchée entre le système d'inférence flou et les réseaux de neurones.

III.3.1 Réseau neuro-flou concurrent :

Dans un système Neuro-flou concurrent, RNA aide le SIF continûment pour déterminer les paramètres exigés particulièrement si les variables d'entrée du contrôleur ne peuvent pas être mesurées directement. Dans certains cas les sorties de SIF ne pourraient pas être directement applicables au processus.

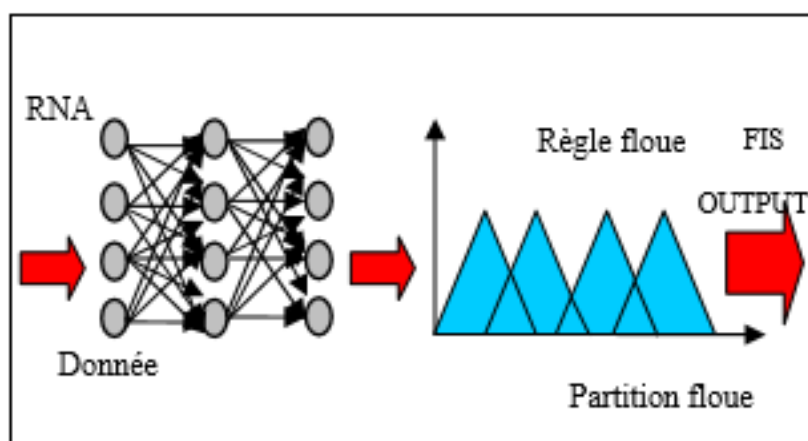


Figure III.3 Système neuro-flou concurrent

III.3.2 Réseau neuro-flou coopératifs :

Un système Neuro-flou coopératif peut être considéré comme préprocesseur où le mécanisme d'apprentissage de réseaux de neurones artificiels (RNA) détermine les fonctions d'appartenance de Système d'inférence flou (SIF) ou les règles floues à partir données d'apprentissage. Une fois que les paramètres de SIF sont déterminés, RNA va au fond. Les fonctions d'appartenance sont habituellement approximer à partir RNA par les données d'apprentissage. [14]

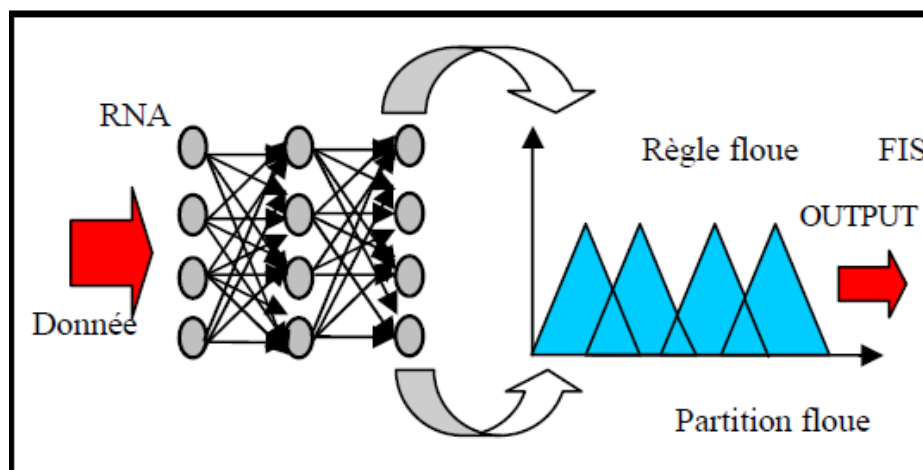


Figure III.4 : Système neuro-flou Coopératif

III.3.4 Les systèmes neuro- flous hybrides :

Les approches neuro-floues modernes sont de cette forme. Un réseau neuronal et un système flou sont combinés dans une architecture homogène. Le système peut être interprété comme un réseau neuronal spécial avec des paramètres flous ou comme un système flou mis en application sous une forme distribuée parallèle.

III.4 Types des systèmes neuro flous

Diverses associations des méthodes et architectures neuro-floues ont été développées depuis 1988. La suivante figure montre quelques modèles des systèmes

- **FALCON (Fuzzy Adaptive learning Control Network)**

Il s'agit de modèles à 5 couches, utilisant la fuzzification en entrée et la défuzzification en sortie. Ceci correspond à l'interprétation juste de la technique de Mamdani. La précision accrue des résultats provoque une lenteur dans l'exécution du système. Ce modèle est rarement utilisé en pratique mais il reste le meilleur pour la commande.

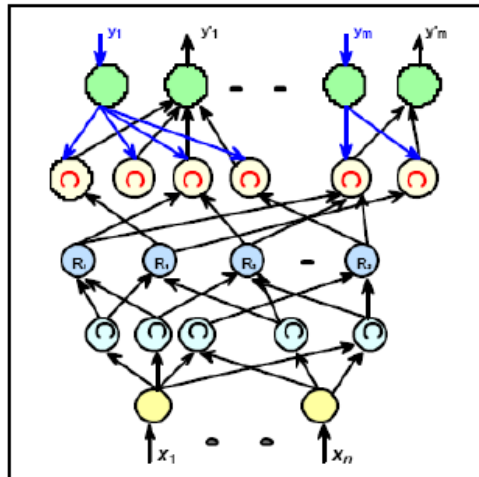


Figure III .5 Architecture de FALCON

- **NEFCLASS**

Modèle utilisé généralement en classification, il est constitué de 3 couches : Une couche d'entrée avec les fonctions d'appartenance, une couche cachée représentée par des règles et une couche de sortie définissant les classes. Ce modèle est facile à mettre en application, il évite l'étape de défuzzification, tout en étant précis dans le résultat final, avec une rapidité bien supérieure aux autres architectures types.

- **ANFIS (Adaptative Network-based Fuzzy Inference System)**

ANFIS représente un système à inférence floue mis en application dans le cadre des réseaux adaptatifs. Ce sont des systèmes hybrides utilisant l'inférence floue de TakagiSugeno. Cette architecture affine les règles floues obtenues par des experts humains pour décrire le comportement d'entrée-sortie d'un système complexe. Ce modèle donne de très bons résultats en poursuite de trajectoire, approximation non linéaire, commande dynamique et traitement du signal.

III.5 Le modèle ANFIS

III.5.1 Architecture de l'ANFIS :

Le réseau adaptatif ANFIS est un réseau multicouche à cinq couches dont les connexions ont toutes un poids égal à 1. Les nœuds sont de deux types différents selon leurs fonctionnalités. Les nœuds carrés (adaptatifs) contiennent des paramètres, et les nœuds circulaires (fixes) n'ont pas de paramètres. Toutefois, chaque nœud applique une fonction sur ses signaux d'entrée. Pour la simplicité, nous supposons que le système d'inférence floue à

Chapitre III : Systèmes neuro-flous

deux entrées x et y , et z comme une sortie. Supposer que la base de règle contient deux règles floues de type Takagi-Sugeno.

Règle1 : $Si x_1 \text{ est } A_1 \text{ et } x_2 \text{ est } B_1 \text{ ALORS } z_1 = a_1 x_1 + b_1 x_2 + c_1$ III.1

Règle2 : $Si x_1 \text{ est } A_2 \text{ et } x_2 \text{ est } B_2 \text{ ALORS } z_2 = a_2 x_1 + b_2 x_2 + c_2$ III.2

Jang a proposé la représentation de cette base de règles par le réseau adaptatif de la figure suivante :

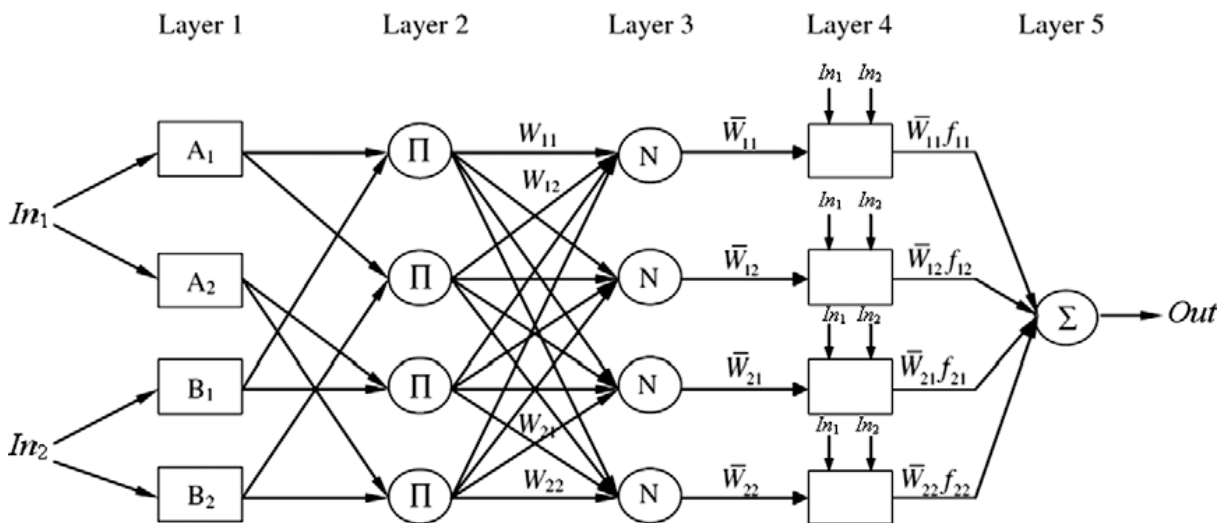


Figure III.6 : L'Architecture de l'ANFIS.

La sortie O_i^k du nœud i de la couche k , dit (nœud (i, k)) dépend des signaux provenant de la couche $k-1$ et des paramètres du nœud (i, k) , c'est-à-dire :

$$O_i^k = f(O_1^{k-1}, \dots, O_{n_{k-1}}^{k-1}, a, b, c \dots)$$

Ou n_{k-1} est le nombre de nœuds dans la couche $k-1$ et a, b, c sont les paramètres du nœud (i, k) pour un nœud circulaires ces paramètres sont nuls.

- **Couche 1 : fuzzification**

Cette couche permet la "fuzzification" des entrées. Chaque neurone dans cette couche correspond à une variable linguistique. Les entrées sont passées par la fuzzification en utilisant des fonctions d'appartenance qui sont généralement sous formes triangulaires, trapézoïdales ou Gaussiennes.

$$O_i^l = \mu_{A_i}(x)$$

X : entrée du nœud.

A_i : terme linguistique associée à sa fonction.

μ : degré d'appartenance de X à A_i.

O : sortie du nœud i.

- **Couche 2 : règles**

Chaque nœud correspond à une T-Norme floue (l'opérateur T-Norme permet de réaliser l'équivalent d'un "ET" booléen). Il reçoit la sortie des nœuds de fuzzification et calcule sa valeur de sortie grâce à l'opérateur produit (Cet opérateur est généralement utilisé mais il en existe d'autres : max, min ...).

$$W_{ij} = \mu_{A_i}(x_1) * \mu_{B_j}(x_2) \text{III.3}$$

i : représente le nombre de partition x_1

j : représente le nombre de partition x_2 .

- **Couche 3 : Normalisation**

Cette couche normalise les résultats fournis par la couche précédente. Les résultats obtenus représentent le degré d'implication de la valeur dans le résultat final.

$$V_{ij} = \frac{W_{ij}}{\sum_{i=1}^{2,2} \sum_{j=1} W_{ij}} \text{III.4}$$

- **Couche 4 : calcul de la sortie des règles**

Chaque nœud de cette couche est relié aux entrées initiales. On calcule le résultat en fonction de son entrée et d'une combinaison linéaire du premier ordre des entrées initiales (Approche de TAKAGI - SUGENO).

$$O_1^4 = V_{ij} f_{ij} = V_{ij} (a_{ij}x_1 + b_{ij}x_2 + c_{ij}) \text{III.5}$$

Où V_{ij} est la sortie de la couche 3, et $\{a_{ij}, b_{ij}, c_{ij}\}$ est l'ensemble des paramètres de sortie de la règle (i,j).

- **Couche 5: Sommation**

Comprend un seul neurone qui fournit la sortie de ANFIS en calculant la somme des sorties de tous les neurones de défuzzification.

$$O_1^5 = Y = \sum V_{ij} f_{ij} \text{III.6}$$

Chapitre III : Systèmes neuro-flous

La figure suivante représente un système ANFIS, à 2 entrées chaque entrée repartie en trois sous ensemble flou et 9 règles.

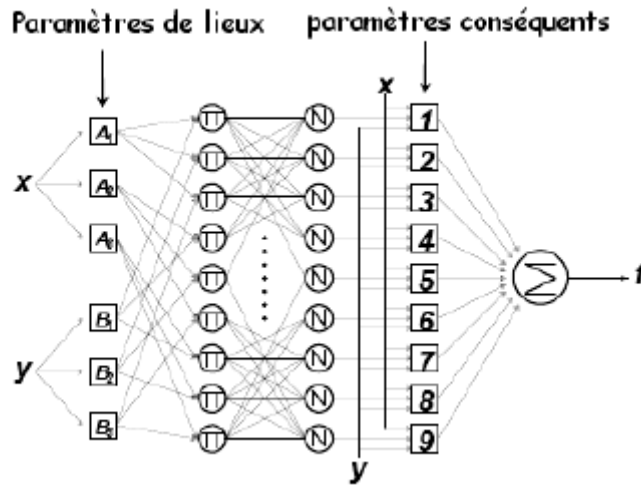


Figure III.7 : Exemple ANFIS à 2 entées avec 9 règles.

Les différentes couches	Types de couches	Le nombre de neurone dans la couche
Couche 0	Les entrées	N
Couche 1	Les valeurs	(p,n)
Couche 2	Les règles	p^n
Couche 3	La normalisation	p^n
Couche 4	Linéarisation des fonctions	p^n
Couche 5	Somme	1

Tableau III.1 : les différentes couches d'un système ANFIS

Tel que :

n : le nombre des entrées.

p : le nombre des sous-ensembles flous d'entrée (partition flou).

III.5.2 Apprentissage du réseau ANFIS :

Pour estimer les paramètres du réseau de neurones, nous avons utilisé le filtre de KALMAN étendu comme un algorithme d'apprentissage. Le régulateur est caractérisé par un vecteur de paramètres \emptyset . Notre objectif est de trouver les valeurs du vecteur \emptyset en minimisant l'erreur suivante:

$$e(k) = \frac{1}{2}(y_d(k) - y(k))^2 \quad k=1, \dots, 2, \quad \text{III.7}$$

Chapitre III : Systèmes neuro-flous

L'approche du filtre de KALMAN étendu consiste à linéariser à tout instant la sortie y autour du vecteur estimé $\hat{\phi}$. Ceci revient à écrire :

$$y_d(k) = y(k) + \psi_i^T(k)(\phi_i - \hat{\phi}_{i-1}) \quad \text{III.8}$$

$$\psi_i^T(k) = \frac{\partial y(k)}{\partial \phi_i} \quad \text{III.9}$$

Par conséquent, les paramètres sont ajustés selon la relation suivante :

$$\phi_i = \phi_{i-1} + P_i \psi_i^T e_i \quad \text{III.10}$$

Avec

$$\left\{ \begin{array}{l} e = y_d - y \\ \psi_i^T = [0_{3,i} \quad 0_{3,i} e \quad 0_{3,i} \dot{e}] \\ P_i = \frac{\alpha_1}{\psi_2 + \psi_i^T \psi_i} \end{array} \right. \quad \text{III.11}$$

$p(k)$ est le gain de l'algorithme d'estimation, α_1 et α_2 fixés à 1 sont des gains d'adaptation pour modifier la vitesse de convergence.

Nous remarquons que cette méthode nécessite le calcul du gradient qui n'est rien d'autre que la méthode de rétro propagation utilisée dans les réseaux de neuronaux [16].

$$\frac{\partial y}{\partial \phi_i} \quad \text{III.12}$$

Pour le cas étudié, le vecteur des paramètres est:

$$\phi_i = [a_i b_i c_i]^T$$

Par conséquent nous avons :

$$\psi_i = \frac{\partial y}{\partial \phi_i} = \left[\frac{\partial y}{\partial a_i} \quad \frac{\partial y}{\partial b_i} \quad \frac{\partial y}{\partial c_i} \right] \quad \text{III.13}$$

III.5.3 Les avantages de l'ANFIS

Les avantages de cette technique stratégique sont :

- Exploitation de la connaissance disponible, grâce à la base de règles.

- Réduction de la taille de la base de règles : il suffit d'avoir des règles générales, les détails seront fournis par le RN.
- Réduction de la complexité de l'apprentissage : le RN doit simplement apprendre les cas particuliers ou les exceptions, pas le problème complet.
- Efficacité immédiate dès le début de l'apprentissage et possibilité d'éviter des comportements initiaux erratiques.

III.6 Conclusion

Dans ce chapitre nous avons présentées le concept de la commande neuro floue de type ANFIS, Cette commande permet d'intégrer des connaissances partielles issues de l'expertise et de données. La connaissance experte est utilisée avec les relations causales dans une première modélisation assez simple et grossière mais nécessaire. La connaissance experte est ensuite exprimée sous formes de règles floues et de contraintes sur la fuzzification des entrées, alors que la phase d'apprentissage permet d'ajuster les paramètres non définis à l'aide de données.

L'utilisation de connaissances expertes permet ainsi de résoudre en partie les problèmes d'exhaustivité des comportements représentés par un ensemble de données. En revanche, les capacités d'apprentissage de l'ANFIS permettent de combler le manque de précision issue de l'expertise. C'est donc un système gagnant où les deux sources de connaissance (règles et données) permettent de combler des lacunes de l'autre.

IV.1. Introduction

Les performances d'un avion dépendent grandement des propriétés physiques (densité, température, pression) de l'air dans lequel il vole. Avant d'entamer l'étude des performances et les forces qui s'exercent sur les avions, il convient d'en donner une idée globale sur la géométrie d'un avion et de définir des repères permettant de situer l'avion par rapport aux différents trièdres. On s'intéresse donc dans ce chapitre aux différentes forces et moments appliqués à l'aéronef [16].

IV.2 Description de la dynamique du vol :

Un avion est un système dynamique complexe, difficile à aborder dans toute sa généralité. L'appareil est un solide déformable, comportant des pièces tournantes (hélices, réacteurs) ou mobiles (gouvernes). Il est soumis à des forces externes très variables, dépendant à la fois de sa trajectoire, de son attitude, de ses déformations et de l'action de la pesanteur terrestre.

Le modèle de l'avion peut être exprimé par six degrés de liberté comme le montre la figure suivante :

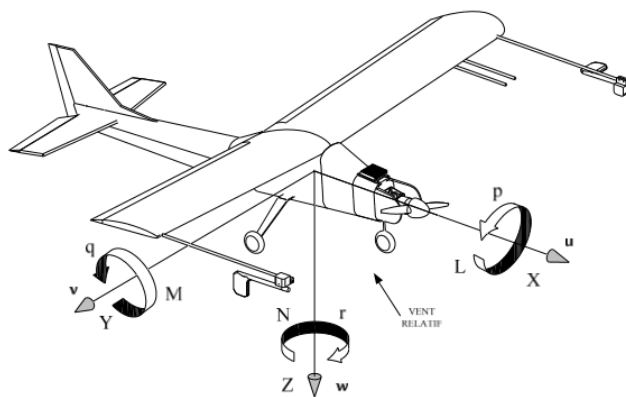


Figure IV.1 Représentation des six degrés de liberté d'un avion

Le tableau IV.2 montre les différentes variables utilisées pour la modélisation :

Chapitre IV : Etude de la dynamique du vol

Angle de navigation	Angle de montée	γ
	Angle de cap	χ
Angles	Angle d'inclinaison	μ
	Angle d'attaque	α
	Angle de dérapage	β
Vitesses angulaires	Vitesse de roulis	P
	Vitesse de tangage	Q
	Vitesse de lacet	R
Moments	Moment de roulis	L
	Moment de tangage	M
	Moment de lacet	N
Coefficients Aérodynamiques	Coefficient de roulis	C_L
	Coefficient de tangage	C_M
	Coefficient de lacet	C_N
Angles de déflexion	Ailerons	δ_a
	Gouverne de profondeur	δ_e
	Gouverne de direction	δ_r

Tableau IV.1. Variable mise en jeu lors de la modélisation

IV .3 Géométrie générale d'un avion :

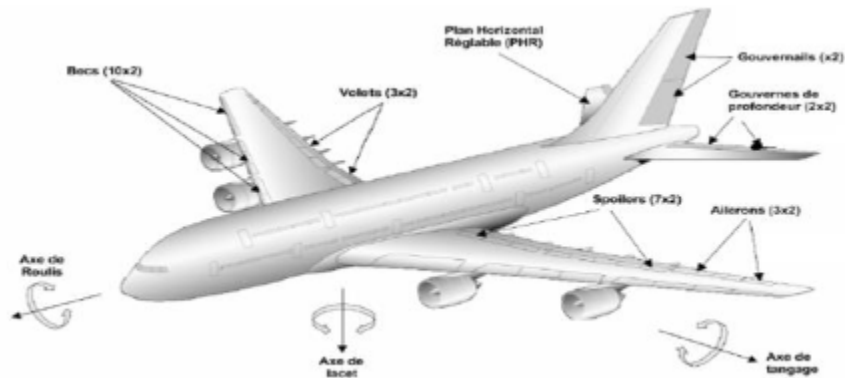


Figure IV.2 : les composants d'un avion

IV.3.1. Fuselage :

Le fuselage désigne l'enveloppe d'un avion qui reçoit généralement la charge transportée, ainsi que l'équipage. La forme d'un fuselage est étudiée pour réduire sa traînée aérodynamique, d'où son nom, issu du verbe fuseler qui signifie littéralement rendre étroit.

IV.3.2. L'aile

Le choix des ailes impose le choix de la forme, des dimensions et de la position. Ces choix vont permettre de compléter la stabilité et gérer la vitesse de l'avion.

L'aile comporte différentes gouvernes, elles sont soumises à des contraintes importantes comme le montre la figure ci-dessous :

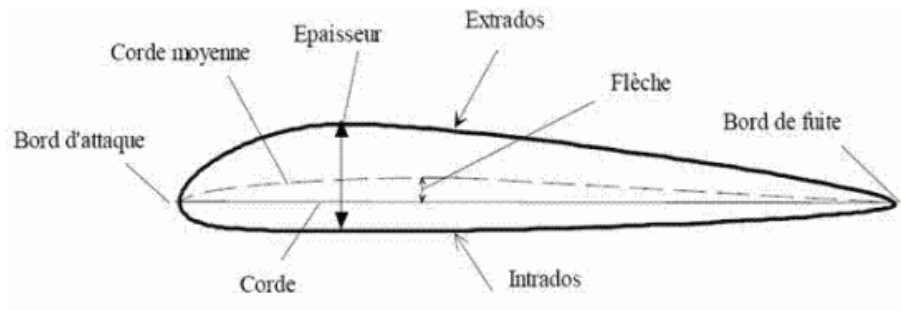


Figure IV.3 : Géométrie de l'aile

IV.3.3. Les gouverne :

Une gouverne est une surface mobile agissant dans un fluide (air ou eau) et servant à piloter un mobile selon trois axes comme illustre dans la figure ci-dessous :

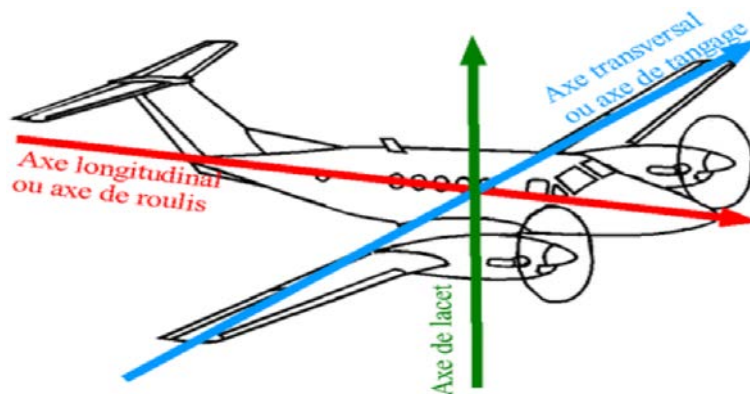


Figure IV.4 : Représentation des trois principaux axes

a) Axe de roulis :

Le roulis est un mouvement de rotation horizontal d'un mobile autour de son axe longitudinal (axe de roulis) qui est contrôlé par la commande des ailerons, qui sont des surfaces mobiles situées au bord de fuite de l'aile.

b) Axe de lacet :

Le lacet est un mouvement de rotation vertical d'un mobile commandé par l'action sur les palonniers, qui sont des dispositifs destinés à actionner la gouverne de direction d'un avion, qui est articulée à l'arrière de la dérivée et assure le mouvement de l'avion.

c) **Axe de tangage :**

Le tangage est un mouvement de rotation autour de l'axe transversal d'un mobile, il est contrôlé par l'empennage horizontal, qui est situé sur la partie arrière du fuselage son rôle est d'assurer la stabilité (partie fixe) et la maniabilité (partie mobile).

IV.4. Les repères de référence :

Les forces et les moments qui s'exercent sur un avion doivent être exprimés dans un repère convenablement défini. L'aérodynamique et la mécanique du vol font appel à différents types de repères aussi appelés trièdres et il convient d'y accorder une attention particulière [17].

IV.4.1. Trièdre terrestre $R_0(OX_0Y_0Z_0)$:

La position dans l'espace d'un objet mobile est a priori définie dans un repère fixe, c'est-à-dire liée à la terre.

L'axe X_0 est dirigé suivant une référence d'azimut (nord vrai ou nord magnétique en général). L'axe Z_0 est orienté dans la direction du centre de la terre. L'axe Y_0 est perpendiculaire au plan (X_0, Z_0) et est orienté vers l'Est.

IV.4.2. Trièdre aérodynamique $R_s(OX_sY_sZ_s)$:

C'est le repère de stabilité, l'axe X_s est orienté suivant la direction du vecteur de vitesse air. L'axe Z_s est perpendiculaire à l'axe X_s et orienté vers le bas. L'axe Y_s est perpendiculaire au plan (X_s, Z_s) et est orienté vers la droite.

IV.4.3. Trièdre avion $R_a(OX_aY_aZ_a)$:

Ce trièdre est une base d'étude pour le mouvement de l'avion par rapport aux autres trièdres de référence. L'axe X_a est issu de l'origine O vers le nez de l'avion, c'est l'axe longitudinal de l'avion. L'axe Y_a est perpendiculaire au plan de symétrie de l'avion et est orienté vers la droite. Le plan (X_a, Z_a) coïncide avec le plan symétrique de l'avion. L'axe Z_a est perpendiculaire au plan horizontal de l'avion (X_a, Y_a) et qui est orienté vers le bas. Ce repère sera utilisé pour étudier le mouvement de l'avion.

IV.5. Passage entre repères :

Un passage entre un repère est un autre peut s'effectuer en prenons par exemple le passage du trièdre avion R_a vers le trièdre terrestre R_0 Porté par l'avion La position du trièdre avion $R_a(OX_aY_aZ_a)$ peut être repérée par rapport au trièdre terrestre $R_0(OX_0Y_0Z_0)$ par les angles d'Euler. Soit le trièdre auxiliaire $OX'Y'Z'$ confondu avec le trièdre normal terrestre qui est amené en coïncidence avec le trièdre avion à l'aide de trois rotations successives.

La matrice de passage correspond à trois rotations du trièdre terrestre local au trièdre avion. Une première rotation d'angle ψ autour de l'axe Z_0 ensuite une deuxième rotation d'angle θ et enfin une dernière rotation d'angle φ . Les angles d'Euler ψ , θ et φ portent respectivement les noms d'azimut, d'assiette longitudinale et d'angle de gîte.

$$\begin{pmatrix} X_a \\ Y_a \\ Z_a \end{pmatrix} = E \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix}$$

$$E = R_\varphi \cdot R_\theta \cdot R_\psi$$

L'expression de la matrice E en fonction des angles d'Euler est :

$$E = \begin{pmatrix} \cos\psi\cos\theta & \sin\psi\cos\theta & -\sin\theta \\ \sin\psi\cos\varphi + \cos\psi\sin\theta\sin\varphi & \cos\psi\cos\varphi + \sin\psi\sin\theta\sin\varphi & \cos\theta\cos\varphi \\ \sin\psi\sin\theta\cos\varphi + \sin\psi\sin\varphi & -\cos\psi\sin\varphi + \sin\psi\sin\theta\sin\varphi & \cos\theta\cos\varphi \end{pmatrix}$$

IV.6 Les forces appliquées sur un avion

La physique d'un avion en vol peut se résumer par l'équilibre des 4 forces opposables deux à deux comme le montre l'image ci-dessous :

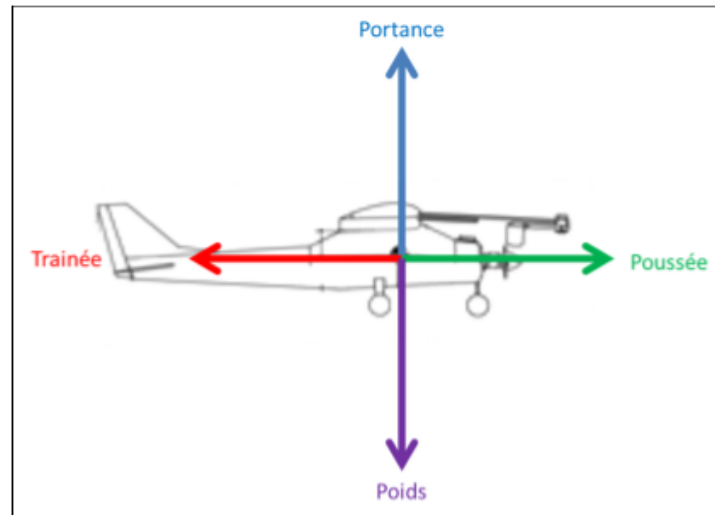


Figure IV.5 : Forces subies par l'avion

Ces quatre forces agissent deux par deux. la portance (toutes les forces qui s'exercent vers le haut) s'oppose au poids (toutes les forces qui s'exercent vers le bas), et la poussée s'oppose à la trainée, ces quatre forces agissent en un point unique appelé centre de gravité.

1. Le poids

Le poids est la force dirigée verticalement vers le bas, en autre terme le poids s'exerce toujours selon une ligne qui va du centre de gravité de l'avion vers le centre de la terre. Le poids change seulement en fonction de la consommation du carburant [18].

2. La portance

Lorsque l'air s'écoule sur l'aile, il y a création de forces aérodynamiques. La portance est la distribution de la pression sur la surface, elle s'applique sur toute l'aile, Sa résultante est une force verticale vers le haut.

3. La poussée

La poussée est exercée par les réacteurs qui repoussent l'air vers l'arrière. Ainsi par la loi action réaction l'air pousse l'avion vers l'avant. Donc, la poussée fournie par le ou les moteurs d'un avion sert à le propulser dans l'air. La poussée s'oppose à la trainée, ces deux forces s'équilibrent lors d'un vol stabilisé.

A même amplitude si la poussée augmente, cette dernière dépasse momentanément la trainée et l'avion prend de la vitesse. Mais la trainée augmente aussi dans ce cas et rattrape la

poussée au bout d'un certain temps. L'avion cesse alors d'accélérer et reprend un vol stabilisé.

4. La trainée

La trainée est la force qui s'oppose au mouvement de l'avion dans l'air. C'est une force qui s'applique sur toute l'aile.

IV.7 Équations de mouvement

Les équations générales du mouvement sont données par un ensemble des équations non linéaire, décrites dans ce qui suit. Les équations des forces et des moments sont dans les expressions (de IV.2 a IV.6 respectivement), elles sont réarrangées dans une configuration d'espace d'états non-linéaire, telles que les vitesses linéaire et angulaires sont des variables d'état, les forces externes et les moments sont des entrées de l'espace d'état. En plus des vitesses linéaire et angulaires, les angles d'Euler et les coordonnées de positions par rapport a la terre sont ajoutées pour compléter le vecteur d'état [4].

Le lecteur peut se référer à la version plus détaillée de Nelson (1998) sur la dynamique de l'aéronef. En effet, dans son ouvrage, Nelson présente un ensemble plus général des équations de forces et de moment à partir des équations pour une masse ponctuelle.

Le modèle d'état de la dynamique du vol peut être exprimé comme suit :

$$\begin{cases} \dot{\vec{\chi}} = F(\vec{X}, \vec{U}, \vec{d}, t) \\ \vec{y} = C\vec{X} \end{cases} \text{IV.1}$$

Avec

F est fonction de vecteurs multi variables

$\vec{\chi}$ Représente le vecteur d'état avec les composantes suivantes

$$\vec{\chi} = [V \alpha \beta p q r \psi \theta \varphi x y h]$$

$\vec{X} = [x y h]^T$ Les composantes du vecteur définissent la position de l'avion par rapport à la terre, ou h est l'altitude.

$\vec{U} = [\delta_e \delta_{th} \delta_a \delta_r]^T$ Est le vecteur de commande ou δ_e est la déflexion de la gouverne de profondeur, δ_{th} est l'accélération du moteur, δ_a est la déflexion de l'aileron et δ_r et la déflexion des gouvernes de directions.

Chapitre IV : Etude de la dynamique du vol

$\vec{d} = [V_{gx} V_{gy} V_{gz}]^T$ Est le vecteur de perturbations qui représente les projections de la vitesse instantanée de la turbulence dans les axes de l'avion.

C : Est la matrice d'observation.

- **Les équations des forces :**

Les équations des forces sont déterminées à partir de la deuxième loi de Newton

$$\sum F_{ex} = m \frac{dV}{dt}$$

On obtient :

$$X(\vec{U}, \vec{X}, \vec{d}) = mg \sin \theta + m(\dot{u} + qw - rv)$$

$$Y(\vec{U}, \vec{X}, \vec{d}) = -mg \cos \theta \sin \varphi + m(\dot{v} + ru - pw) \quad \text{IV.2}$$

$$Z(\vec{U}, \vec{X}, \vec{d}) = -mg \cos \theta \cos \varphi + m(\dot{w} + pv - qu)$$

En faisant un réarrangement pour ces équations aérodynamiques des forces propulsives on aura les équations du mouvement linéaires suivantes :

$$\dot{u} = rv - qw - g \sin \theta + \frac{1}{m} X$$

$$\dot{v} = pw - ru + g \sin \theta \cos \varphi + \frac{1}{m} Y \quad \text{IV.3}$$

$$\dot{w} = qu - pv + g \cos \theta \cos \varphi + \frac{1}{m} Z$$

- **Les équations des moments :**

Les équations des moments sont déterminées à partir de la deuxième loi de Newton

$$\sum M_{ex} = I \frac{d\omega}{dt}$$

ω : La vitesse angulaire

I : matrice d'inertie donnée comme suit :

$$I = \begin{pmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{pmatrix}$$

On obtient donc :

$$L(\vec{U}, \vec{X}, \vec{d}) = I_{xx}\dot{p} - (I_{yy} - I_{zz})qr - I_{yz}(q^2 - r^2) - I_{zx}(\dot{r} + pq) - I_{xy}(\dot{q} - rp)$$

$$M(\vec{U}, \vec{X}, \vec{d}) = I_{yy}\dot{q} - (I_{zz} - I_{xx})rp - I_{zx}(r^2 - p^2) - I_{xy}(\dot{p} + qr) - I_{yz}(\dot{r} - pq) \text{IV.4}$$

$$N(\vec{U}, \vec{X}, \vec{d}) = I_{zz}\dot{r} - (I_{xx} - I_{yy})pq - I_{xy}(p^2 - q^2) - I_{yz}(\dot{q} + rp) - I_{zx}(\dot{p} - qr)$$

- **Les équations d'Euler :**

$$\dot{\theta} = q\cos\varphi - r\sin\varphi$$

$$\dot{\psi} = q\sin\varphi\sec\theta + r\cos\varphi\sec\theta \text{IV.5}$$

$$\dot{\phi} = p + q\sin\varphi\tan\theta + r\cos\varphi\tan\theta$$

- **Equations de vitesse par rapport au sol :**

La vitesse d'un avion par rapport au repère géocentrique est donnée par les trois composantes vectorielles suivantes :

Selon l'axe X :

$$\dot{X} = u\cos\Psi\cos\theta + v(-\sin\psi\cos\varphi + \cos\psi\sin\theta\sin\varphi) + w(\sin\psi\cos\varphi + \cos\psi\sin\theta\cos\varphi) \text{IV.6}$$

Selon l'axe Y :

$$\dot{Y} = u\sin\psi\cos\theta + v(\cos\psi\cos\varphi + \sin\psi\sin\theta\sin\varphi) + w(-\cos\psi\sin\varphi + \sin\psi\sin\theta\cos\varphi) \text{IV.7}$$

Selon Z (altitude = -Z)

$$\dot{h} = u\sin\theta - (v\sin\varphi + w\cos\varphi)\cos\theta \text{IV.8}$$

IV.8 La théorie des petites perturbations

En utilisant la théorie des petites perturbations pour linéariser le modèle obtenu précédemment. En appliquant cette théorie, on assume que le mouvement de l'avion consiste à des petites perturbations concernant les conditions de vol (petites déviations autour de son point de fonctionnement). Toutes les variables dans les équations du mouvement sont remplacées par des termes d'équilibre et de perturbation :

$$u = u_0 + \Delta u ; v = v_0 + \Delta v ; w = w_0 + \Delta w ; p = p_0 + \Delta p ; q = q_0 + \Delta q$$

$$r = r_0 + \Delta r ; X = X_0 + \Delta X ; Y = Y_0 + \Delta Y ; Z = Z_0 + \Delta Z$$

$$M = M_0 + \Delta M ; N = N_0 + \Delta N ; L = L_0 + \Delta L ; \theta = \theta_0 + \Delta \theta ; \delta = \delta_0 + \Delta \delta$$

IV.9 Linéarisation de la dynamique du vol

Les équations régissant le mouvement de l'avion sont non linéaires, cela revient à la présence du produit entre deux variables, puissances supérieures à un et d'autres fonctions non linéaires. Certaines simplifications peuvent se faire autour d'un point de fonctionnement, cependant, les variables seront construites à partir de deux autres, comme le montre *la théorie des petites perturbations*.

Chaque équation régissant du mouvement de l'avion sera linéarisé autour du point de fonctionnement. En négligeant le produit de deux grandeurs en minuscules et en considérant les transformations trigonométriques suivantes : $\sin \varepsilon \approx \varepsilon$ et $\cos \varepsilon \approx 1$ pour des petites valeurs de ε .

- **Equations linéarisées de la vitesse du véhicule, de l'angle d'attaque alpha et de l'angle de dérapage**

Dans le domaine des véhicules aériens, la contribution aux forces et moments sont à venir surtout de l'aérodynamique des ailes, le corps et l'empennage. Il serait difficile d'exprimer ceci en fonction des variables de mouvement du véhicule u , v et w . Cependant, il est beaucoup plus facile de les formuler en fonction de la vitesse du véhicule V , l'angle d'attaque α et l'angle de dérapage β . On peut exprimer les vitesses linéaires directement en fonction de V , α , β par les relations :

$$u = V \cos \beta \cos \alpha$$

$$v = V \sin \beta \tag{IV.9}$$

$$w = V \cos \beta \sin \alpha$$

A partir des équations de mouvement (IV.9) on tire les dérivés des variables V , β et α par rapport au temps :

$$\begin{bmatrix} \dot{V} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} \cos \alpha \cos \beta & \sin \beta & \sin \alpha \cos \beta \\ -\frac{\sin \alpha}{V \cos \beta} & 0 & \frac{\cos \alpha}{V \cos \beta} \\ -\frac{\cos \alpha \cos \beta}{V} & \frac{\cos \beta}{V} & -\frac{\sin \alpha \sin \beta}{V} \end{bmatrix} \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix}$$

Chapitre IV :Etude de la dynamique du vol

En substituant les équations (IV.3) dans les équations (IV.9) et après quelques manipulations il vient :

$$\begin{aligned}
 \dot{V} &= -g(\sin\theta\cos\alpha\cos\beta - \cos\theta\sin\phi\sin\beta - \cos\theta\cos\phi\sin\alpha\cos\beta) + \frac{X}{m}\cos\alpha\cos\beta + \\
 &Y/m\sin\beta + Z/m\sin\alpha\cos\beta \\
 \dot{\alpha} &= q - p\cos\alpha\tan\beta - r\sin\alpha\tan\beta + g(\cos\theta\cos\phi\cos\alpha + \sin\theta\sin\alpha)/V\cos\beta - \\
 &X/mV\sin\alpha/V\cos\beta + Z/m\cos\alpha/V\cos\beta \text{IV.10} \\
 \dot{\beta} &= p\sin\alpha - r\cos\alpha + \frac{g}{V(\sin\theta\cos\alpha\sin\beta + \cos\theta\sin\phi\cos\beta - \cos\theta\cos\phi\sin\alpha\sin\beta)} \\
 &\quad - \frac{X}{mV}\cos\alpha\sin\beta + Y/mV\cos\beta - Z/mV\sin\alpha\sin\beta
 \end{aligned}$$

Par la linéarisation des équations (IV.10), on peut obtenir un ensemble d'équations linéarisées en fonction des variables de perturbation :

$$\Delta V, \Delta\alpha, \Delta\beta, \Delta p, \Delta q, \Delta r, \Delta X, \Delta Y, \Delta Z, \Delta\theta, \Delta\Psi \text{ et } \Delta\phi$$

Les équations suivantes donnent le modèle linéarisé des équations non linéaires (IV.10), en négligeant les termes d'ordre supérieur (exemple $\Delta\beta\Delta\alpha \approx 0$, $\Delta V\Delta\alpha \approx 0$, etc.) et invoquant en même temps les petites approximation d'angle ($\cos\Delta\beta \approx 1$, $\sin\Delta\beta \approx \Delta\beta$, etc.). Après quelques manipulations, les équations suivantes du mouvement pour les variables de perturbation ΔV , $\Delta\alpha$ et $\Delta\beta$ sont dérivées :

$$\begin{cases}
 \Delta\dot{V} = -g\cos(\theta_0 - \alpha_0)\Delta\theta + \frac{\cos\alpha_0}{m}\Delta X + \frac{\sin\alpha_0}{m}\Delta Z \\
 \Delta\dot{\alpha} = \Delta q - \frac{g}{V_0}\cos(\theta_0 - \alpha_0)\Delta\theta - \frac{\sin\alpha_0}{mV_0}\Delta X + \frac{\cos\alpha_0}{mV_0}\Delta Z \text{IV.11} \\
 \Delta\dot{\beta} = \sin\alpha_0\Delta p - \cos\alpha_0\Delta r + \frac{g}{V_0}\cos\theta_0\Delta\phi + \frac{1}{mV_0}\Delta Y
 \end{cases}$$

- **Linéarisation des moments :**

Chapitre IV : Etude de la dynamique du vol

Dans cette section les équations des composants d'accélération angulaires sont dérivées des conditions d'équilibre des vitesses angulaires effectuées comme suit :

$$\begin{cases} p = p_0 + \Delta p \\ q = q_0 + \Delta q \\ r = r_0 + \Delta r \end{cases}$$

Et les moments d'inertie :

$$\begin{cases} L = Lp_0 + \Delta L \\ M = Mp_0 + \Delta M \\ N = Np_0 + \Delta N \end{cases}$$

Pour la condition de la montée régulière on a :

$$p_0 = q_0 = r_0 = 0 \text{ et } L_0 = M_0 = N_0 = 0$$

Donc on aura :

$$\begin{aligned} \Delta L &= I_{xx}\Delta\dot{p} - I_{xx}\Delta\dot{r} \\ \Delta M &= I_{yy}\Delta\dot{q} - I_{xy}\Delta\dot{p} - I_{yz}\Delta\dot{r} \\ \Delta N &= I_{zz}\Delta\dot{r} - I_{zx}\Delta\dot{p} \end{aligned}$$

En supposant que l'avion a une symétrie par rapport au repère X donc $I_{xy} = I_{yz} = 0$

Donc :

$$\begin{aligned} \Delta L &= I_{xx}\Delta\dot{p} - I_{xx}\Delta\dot{r} \\ \Delta M &= I_{yy}\Delta\dot{q} \\ \Delta N &= I_{zz}\Delta\dot{r} - I_{zx}\Delta\dot{p} \end{aligned}$$

- **Linéarisation des équations d'angles d'Euler**

A partir des équations (IV.5) et en considérant les conditions d'équilibre définies dans la théorie des petites perturbations on aura :

$$\Delta\dot{\theta}\cos\Delta\varphi + \Delta\dot{\Psi}\cos(\theta_0 + \Delta\theta)\sin\Delta\varphi = \Delta q$$

Comme $\cos \Delta\varphi = 1$ et $\Delta\dot{\Psi} \sin \Delta\varphi = 0$ donc on aura :

$$\Delta\dot{\theta} = \Delta q \text{IV.12}$$

$$\Delta\dot{\Psi} \cos(\theta_0 + \Delta\theta) \cos \Delta\varphi - \Delta\dot{\theta} \sin \Delta\varphi = \Delta r$$

Après simplification on obtient :

$$\Delta\dot{\Psi} = \frac{1}{\cos \theta_0} \Delta r \text{IV.13}$$

$\Delta\dot{\phi} - \sin \theta_0 \Delta\dot{\Psi} = \Delta p$ Après simplification on obtient :

$$\Delta\dot{\phi} = \Delta p + \tan \theta_0 \Delta r \text{IV.14}$$

IV.10 Equations linéarisées du mouvement longitudinal

La dynamique du vol est décrite par neuf équations différentielles ordinaires fortement non linéaires $[V \alpha \beta p q r \psi \theta \varphi]$. La linéarisation de ces équations se fait autour d'un état d'équilibre en faisant de petites variations $[\Delta V \Delta \alpha \Delta \beta \Delta p \Delta q \Delta r \Delta \psi \Delta \theta \Delta \varphi]$. On sépare les équations de base du mouvement en deux ensembles distincts : un ensemble correspond aux variables du mouvement longitudinal $[\Delta V \Delta \alpha \Delta q \Delta \theta]$, et un autre ensemble qui correspond à $[\Delta \beta \Delta p \Delta r \Delta \psi \Delta \varphi]$.

Dans notre cas on s'intéresse au mouvement longitudinal, les équations linéarisées sont données comme suit :

$$\frac{\partial X}{\partial u} u + \frac{\partial X}{\partial \dot{u}} \dot{u} + \frac{\partial X}{\partial w} w + \frac{\partial X}{\partial \dot{w}} \dot{w} + \frac{\partial X}{\partial q} q + \frac{\partial X}{\partial \dot{q}} \dot{q} + \frac{\partial X}{\partial \delta_e} \delta_e + \frac{\partial X}{\partial \dot{\delta}_e} \dot{\delta}_e = m[\dot{u} + W_0 q - (g \cos \theta_0) \theta]$$

$$\frac{\partial Z}{\partial u} u + \frac{\partial Z}{\partial \dot{u}} \dot{u} + \frac{\partial Z}{\partial w} w + \frac{\partial Z}{\partial \dot{w}} \dot{w} + \frac{\partial Z}{\partial q} q + \frac{\partial Z}{\partial \dot{q}} \dot{q} + \frac{\partial Z}{\partial \delta_e} \delta_e + \frac{\partial Z}{\partial \dot{\delta}_e} \dot{\delta}_e = m[\dot{w} - U_0 q + (g \sin \theta_0) \theta]$$

$$\frac{\partial M}{\partial u} u + \frac{\partial M}{\partial \dot{u}} \dot{u} + \frac{\partial M}{\partial w} w + \frac{\partial M}{\partial \dot{w}} \dot{w} + \frac{\partial M}{\partial q} q + \frac{\partial M}{\partial \dot{q}} \dot{q} + \frac{\partial M}{\partial \delta_e} \delta_e + \frac{\partial M}{\partial \dot{\delta}_e} \dot{\delta}_e = I_{yy} \dot{q}$$

$$\dot{\theta} = q$$

Les équations peuvent être réécrites sous la forme :

$$\dot{u} = X_u u + X_w w + W_0 q - (g \cos \theta_0) \theta + X_{\delta_e} \delta_e$$

$$\dot{w} = Z_u u + Z_w w + U_0 q - (g \sin \theta_0) \theta + Z_{\delta_e} \delta_e$$

$$\dot{q} = M_u u + M_w w + M_{\dot{w}} \dot{w} + M_q q + M_{\delta_e} \delta_e$$

$$\dot{\theta} = q$$

D'où le modèle d'état :

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} X_u & X_w & W_0 & -g \sin \theta_0 & 0 \\ Z_u & Z_w & U_0 & -g \sin \theta_0 & 0 \\ M_u & M_w & M_q & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -Z_u & -Z_w & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \\ h \end{bmatrix} + \begin{bmatrix} X_{\delta_e} \\ Z_{\delta_e} \\ M_{\delta_e} + M_{\delta_e} Z_{\delta_e} \\ 0 \\ -Z_{\delta_e} \end{bmatrix} \delta_e$$

IV.11 Conclusion

Dans ce chapitre nous avons rappelés les notations fondamentales concernant la mécanique du vol afin de pouvoir étudier les surfaces de commande, les repères de références, les coefficients aérodynamiques ainsi que les équations de mouvements, ce qui nous un permis de construire un modèle non linéaire dans le but de le linéariser et aboutir au modèle linéaires longitudinal. L'application du contrôleur neuro flou sur ce modèle est l'objet du chapitre V.

V.1 Introduction :

Le but de ce chapitre est de synthétiser un correcteur neuro-flou et ajuster ses paramètres, et cela pour arriver aux valeurs exactes ou approximatives de gains qui permettent d'atteindre les résultats désirés, et de faire une comparaison avec le correcteur PID pour mieux voir les performances du correcteur neuro-flou.

V.2 Schéma bloc du système sur Simulink :

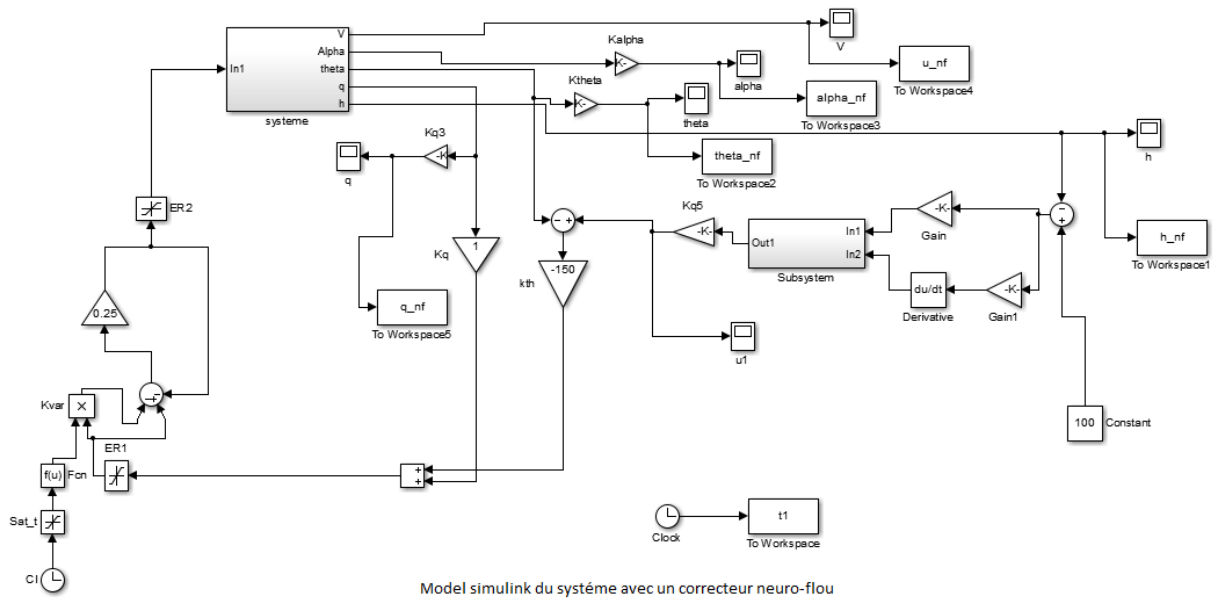


Figure V.1: Model SIMULINK du système.

Le modèle utilisé pendant la simulation est le suivant :

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

Avec :

$$A = \begin{bmatrix} -0.0345 & 6 & -9.78 & 0 & 0 \\ -0.0041 & -1.76 & 0 & 0.99 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0.003 & -25.7 & 0 & -2.19 & 0 \\ 0 & -69.4 & 69.4 & 0 & 0 \end{bmatrix}$$

$$B = [0.36 \ -0.16 \ 0 \ -31.1 \ 0]^T$$

$$C = \text{eye}(5,5)$$

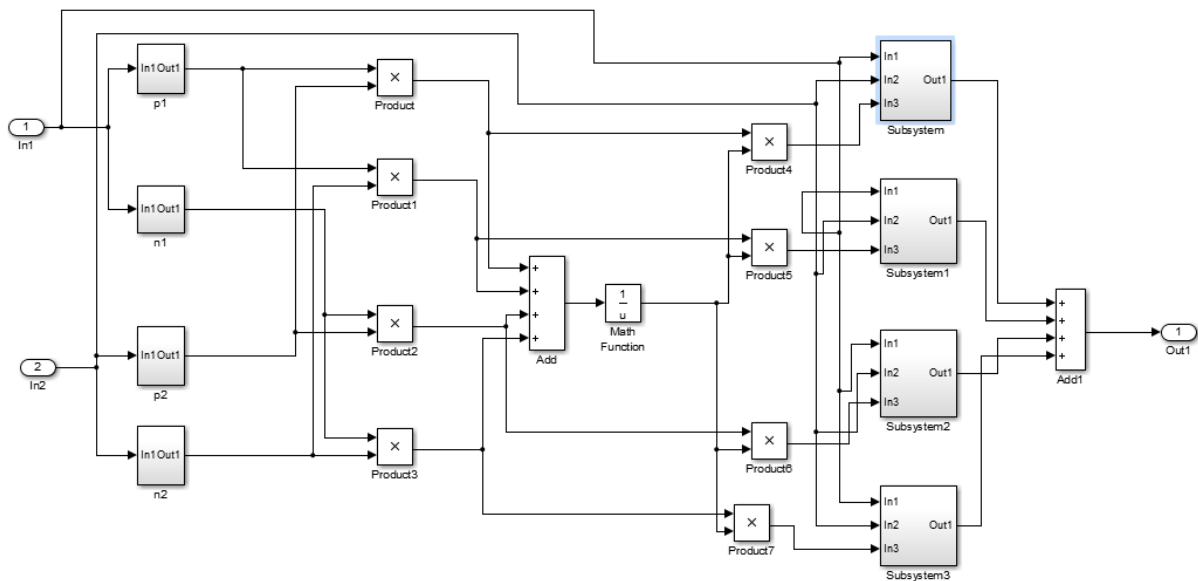


Figure V.2: Model SIMULINK du correcteur neuro flou ANFIS.

Nous avons construit le model SIMULINK du correcteur neuro flou en suivant l'architecture du réseau ANFIS pour deux entrées, et 4 règles.

V.3 Simulation et interprétation des résultats :

A des fins de comparaisons, nous avons fait des simulations pour deux boucles, l'une avec un contrôleur neuro-flou et l'autre avec un contrôleur PID. Et Nous avons obtenus les résultats suivants :

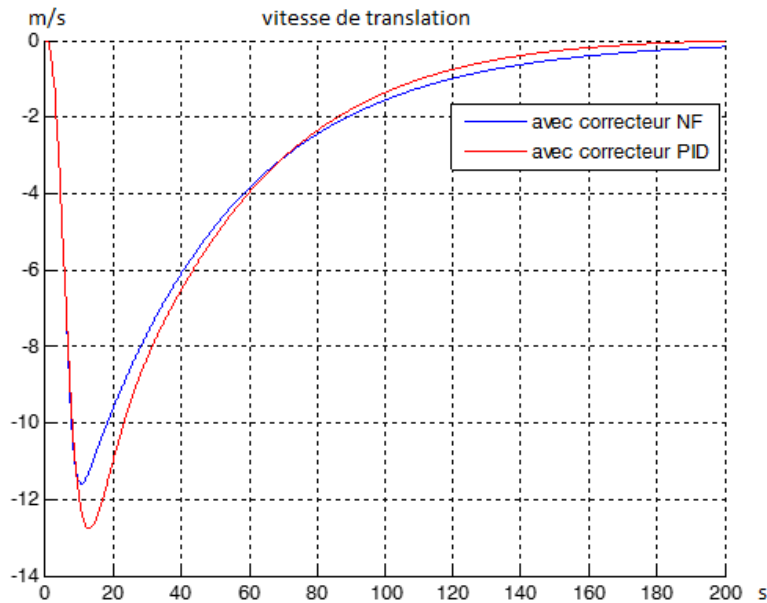


Figure V.3: vitesse de translation en fonction du temps.

On constate que la vitesse de translation pour les deux types de commande diminue à cause de la force générée par l'angle de tangage qui ralentit la translation. On constate aussi que la vitesse de translation diminue avec l'augmentation de l'altitude. Une fois que l'altitude désirée atteinte, la vitesse tend vers zéro c'est-à-dire la stabilisation de l'avion. En comparant les résultats des deux commandes, on remarque que la réponse du système commandé par la commande neuro-floue converge plus rapidement vers le zéro.

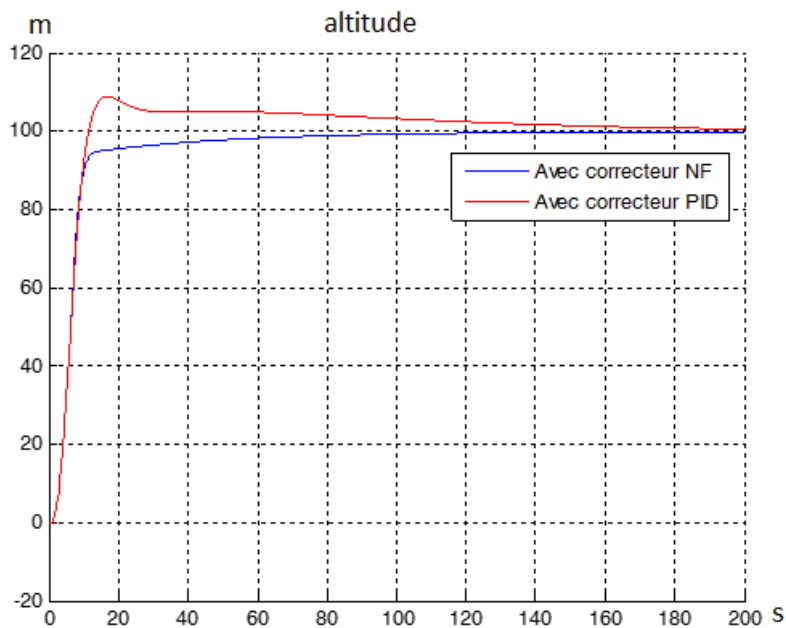


Figure V.4: l'altitude en fonction du temps.

Pour la figure V.6 on observe que la consigne d'altitude est poursuivie par les deux commandes, néanmoins la commande neuro-floue converge progressivement vers la consigne désirée ($h_{ref} = 100$ m) avec un écart statique nul et un temps de réponse à 95s, contrairement à la commande PID ou on constate la présence d'un écart statique et un temps de réponse plus important.

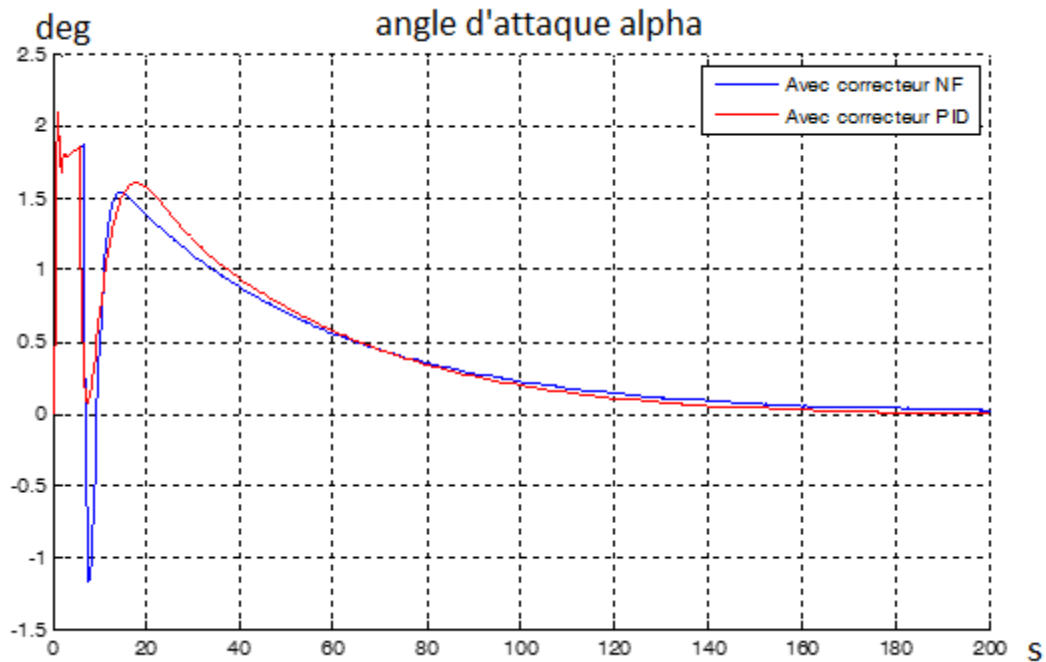


Figure V.5: l'angle d'attaque alpha en fonction du temps.

On remarque que l'angle d'attaque est toujours compris entre $[-6,6]$ afin d'éviter le seuil de décrochage, la consigne est respectée par les deux correcteurs, mais on remarque que la réponse du système commandé par la commande neuro-floue est plus satisfaisante.

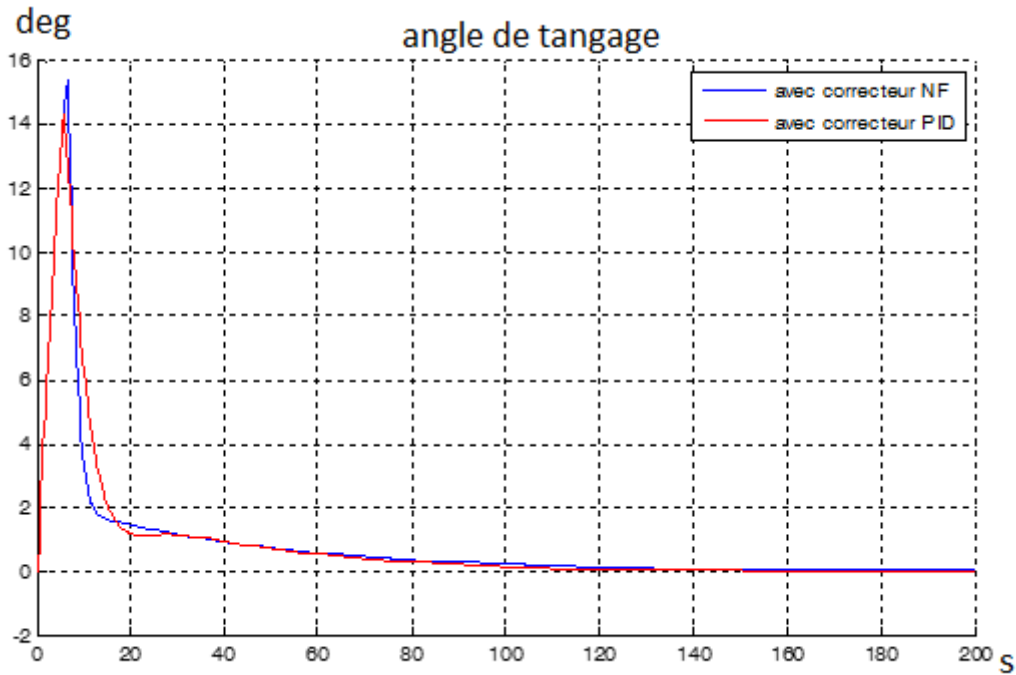


Figure V.6: l'angle de tangage θ en fonction du temps.

Thêta est l'angle de tangage nécessaire pour l'orientation de l'avion vers le bas et le haut pour créer des forces aérodynamiques telle que la portance, il est toujours inférieur à 16 degrés. Pour les deux types de commande l'angle de tangage diminue jusqu'à son annulation lors de l'atteinte de l'amplitude désirée c'est-à-dire la stabilisation de l'avion.

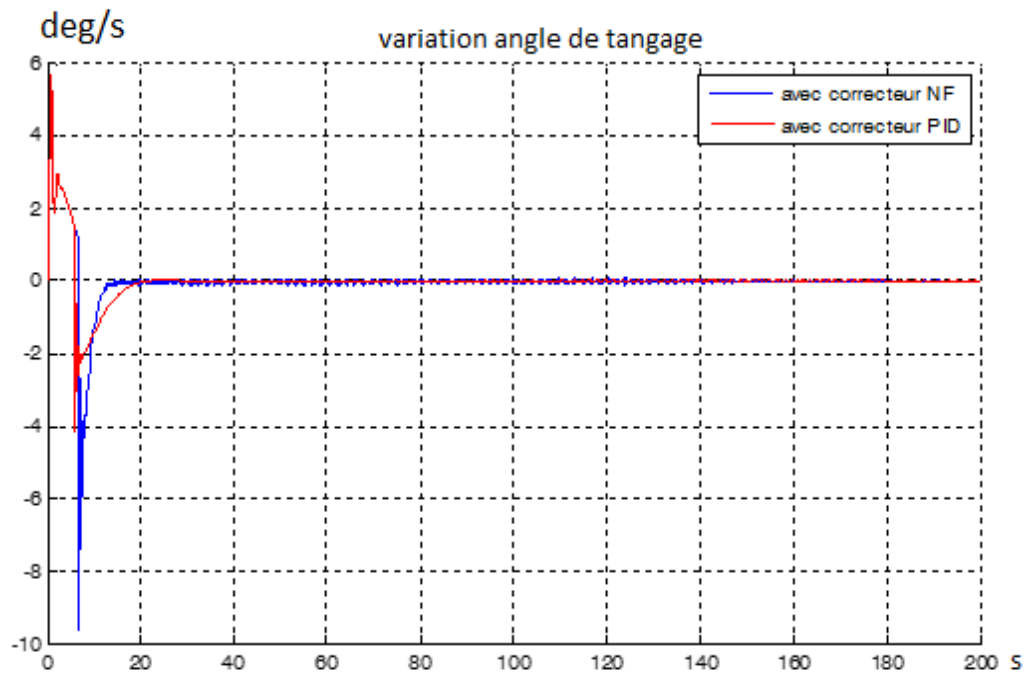


Figure V.7: la variation de l'angle de tangage en fonction du temps.

On observe pour les deux commandes que le taux de variation de tangage augmente en fonction de la variation de l'amplitude qui dépend de l'angle de tangage, et au fur et à mesure que la réponse en altitude s'approche de la consigne le taux de tangage se réduit jusqu'à son annulation.

En vue des résultats obtenus par cette simulation on peut dire que notre contrôleur a pu répondre à nos attentes.

V.4 Conclusion :

Dans ce chapitre nous avons appliqué une commande neuro-floue et une commande PID sur le canal longitudinal d'un drone à voilure fixe afin de pouvoir évaluer et comparer les performances des deux contrôleurs. Les résultats obtenus ont prouvé que les deux contrôleurs répondaient aux attentes. Néanmoins les résultats obtenus avec la commande neuro-flou sont plus satisfaisants.

Conclusion générale :

Conclusion générale :

L'étude présentée dans ce mémoire a pour objectif d'aborder les techniques de l'intelligence artificielle, qui constituent une alternative aux commandes classiques, généralement utilisées en automatique afin de répondre au mieux aux objectifs de performance et de robustesse pour la régulation d'un procédé.

Nous avons décrit une nouvelle architecture de commande non linéaire, l'approche proposée repose sur l'utilisation de la commande neuro-flou basée sur l'architecture ANFIS. Les lois de commande proposées sont appliquées à la stabilisation d'un vol plus précisément la stabilisation du canal longitudinal d'un avion, qui est un système qui possède des non linéarités non négligeable.

Pour y parvenir nous nous sommes amenés à donner quelques définitions de base de la logique floue, qui permet de s'abstenir de l'utilisation des modèles mathématique parfois très complexes et difficiles à obtenir, aussi on a donné une représentation d'un régulateur flou et ses différents constituants.

En second lieu nous avons donné quelques définitions des réseaux de neurones, ses types, ses différentes architectures et enfin les méthodes et algorithmes d'apprentissage.

Puis on s'est intéressé aux systèmes neuro flous qui combine la théorie puissante de la logique floue et celle des réseaux de neurones afin d'introduire des capacités d'apprentissage et d'adaptation dans les systèmes flous, on s'est basé sur l'architecture ANFIS qui est un réseau connexionniste à cinq couches, il est capable d'adapter et d'ajuster ses paramètres en temps réel en utilisant l'algorithme de Kalman.

Ensuite nous avons effectué une étude de la dynamique du vol afin de connaître les constituants du système, son fonctionnement et son modèle mathématique, plus précisément le modèle longitudinal. Afin qu'on puisse implémenter le contrôleur neuro flou à notre système, et comparer par la suite ses résultats de simulations avec un contrôleur PID.

La commande neuro floue a donnée de meilleurs résultats cotés performance, robustesse, poursuite et la rapidité par un meilleur temps de réponse.

Nous concluons que chaque type de commande peut être avantageux dans un sens et désavantageux dans un autre sens, mais on peut affirmer que la commande neuro floue s'avère très efficace pour le contrôle des systèmes complexes.

Bibliographie

- [1] Madaoui M. « Commande floue et neuro-floue d'un convertisseur direct AC/AC monophasé à structure minimale. Application à l'alimentation d'une machine asynchrone monophasé » Thèse de Magister, ENP Alger Février 2007.
- [2] Gerald, H. «processeursflous». *HE-ARC, BAPTISTE SAVOYE 26, CH-2610 STIMIER*, Mai 2008
- [3] Laurant F. Sylvie G. Andre T. " Commande floue 1, de la stabilisation à la supervision " Paris, Hermès Sciences Publication, 2003.
- [4] "Lotfi BAGHLI" Contribution à la commande de la machine asynchrone, utilisation de la logique floue, des réseaux de neurones et des algorithmes génétiques, thèse de doctorat, Faculté des Sciences -54500 Vandœuvre-lès-Nancy, 1999.
- [5] Bernadette bouchon-meunier « la logique floue »
- [6] BUHER.H : réglage par logique floue 1994.
- [7] pierre borne : introduction a la commande floue ; editionteckrip (paris 1998).
- [8] t.takagi, M.sugenofuzzyidentification on systèmes and cybernetics vol 15 ; 1985.
- [9] Borne P. Haggege J. "les réseaux de neurones présentation et applications" Paris : Editions Technip, 2007.
- [10] S.Hay Kin, "Neural Network :ACompréhensive Foundation.Printice.Hall",Inc. :New Jersey,1999.
- [12] H.Chaoui, commande adaptative ce système a dynamique complexe basée sur l'intelligence artificielle ,2013.
- [13] Ould Abdeslam .D "Techniques neuromimétiques pour la commande dans les systèmes électriques : application au filtrage actif parallèle dans les réseaux électriques basse tension" Thèse de Doctorat, université de Haute-Alsace. France. Décembre 2005.
- [14] Mordjaoui M" Modélisation des effets électromagnétiques Apport de la logique floue et Neuro-floue" Thèse de doctorat es-sciences, Université De Batna, Juin 2004.
- [16] Gérard Degrez, « Performance et stabilité des avions », 2001
- [17] BOULANOUAR Khadidja, « Modélisation et Commande d'un Drone », 2016
- [18] Copyright A. Deperrois, « Quelques notions d'aérodynamique de base et leur calcul dans XFLR5 », 2009