

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Ferhat Abbas Sétif - 1

Institut d'Optique et de Mécanique de Précision

THESE

Présentée pour l'obtention du diplôme de

DOCTORAT EN SCIENCES

Option : optique et mécanique de précision

Présentée par :

MR : HARROUCHE FATEH

Thème :

**Etude Comparative de Deux Méthodes D'optimisation d'un Système de
Classification des Défauts Mécaniques par la Logique Floue**

Présentée et soutenue publiquement le 22 / 07 / 2019 devant le jury composé de :

Jury

| | | | | |
|-----------------|-------------------|------------|--------------------|------------|
| HAMOUDA | Abdellatif | Professeur | Université sétif 1 | Président |
| FELKAOUI | Ahmed | Professeur | Université sétif 1 | Rapporteur |
| MELLAH | Rabah | Professeur | UMMTO (Tizi Ouzou) | Examineur |
| HADDAD | Moussa | Professeur | EMP (Alger) | Examineur |

Je dédie ce travail :
A mes chers parents,
A toute ma famille,
A ma femme et mes enfants,
A mes amis.

Remerciements

Cette étude a été effectuée au Laboratoire de Mécanique de Précision Appliquée (LMPA), Institut d'Optique et de Mécanique de précision, Université Ferhat Abbas-Sétif 1(UFAS1).

J'exprime ma profonde gratitude et ma reconnaissance envers Monsieur Felkaoui Ahmed, Professeur à l'Institut D'Optique et Mécanique de Précision, Université Ferhat Abbas Sétif 1, pour l'aide précieuse qu'il m'a apportée dans l'élaboration de ce travail. Je le remercie pour ses conseils, son soutien, sa disponibilité et les discussions fructueuses qu'on a eu ensemble.

Je voudrais adresser mes sincères remerciements à monsieur Abdellatif HAMOUDA, Professeur à l'Institut D'Optique et Mécanique de Précision, Université Ferhat Abbas Sétif 1, d'avoir accepté de présider l'honorable jury de cette thèse.

Je remercie vivement Monsieur Rabah MELLAH, Professeur à L'Université Mouloud Mammeri Tizi Ouzou, et monsieur Moussa Haddad Professeur à l'école Militaire Polytechnique Bordj el Bahri Alger, de m'avoir fait l'honneur d'accepter d'être les Examineur de cette thèse.

Mes remerciements vont également à tous mes collègues ingénieurs, et techniciens de l'institut d'optique et de mécanique de précision, ainsi que les doctorants de ce dernier. Je leurs souhaite à tous du succès dans leurs recherches ainsi que dans leurs vies professionnelles.

Mes remerciements s'adressent finalement à toute ma famille pour sa patience et son encouragement infaillible durant toutes les années de mes études.

Résumé:

La logique floue, par sa caractéristique principale de simulation du raisonnement humain, est classée parmi les techniques de l'intelligence artificielle. Cette technique permet de modéliser, puis de remplacer l'expertise de surveillance et de conduite du processus provenant du concepteur ou de l'utilisateur. La logique floue a contribué dans le développement de plusieurs domaines. En maintenance industrielle, la logique floue est utilisée pour résoudre les problèmes du diagnostic par la classification automatique des signaux vibratoires correspondant aux différents modes de fonctionnement des machines. La technique de la logique floue(LF) souffre d'un ensemble d'inconvénients en phase de modélisation à l'aide des modèles à inférence floue à plusieurs niveaux(le choix et le nombre des variables d'entrée, le choix et le nombre des fonctions d'appartenance, la génération des règles, la méthode de défuzzification, etc). Pour surmonter ces inconvénients, on procède à une optimisation du système flou de diagnostic.

L'objectif de ce travail porte sur l'optimisation d'un système de diagnostic de défauts de roulements à base de la logique floue (système à inférence floue(SIF)). Pour cette raison, nous avons choisi une méthodologie basée sur trois étapes: la première étape consiste à classer les différents défauts à l'aide d'un système initial à inférence floue(SIF), dont les variables d'entrée sont sélectionnées de manière aléatoire à partir d'un ensemble d'indicateurs préparé suite à une étude préalable de l'ensemble des signaux enregistrés à partir du système étudié.

Pour des fins d'optimisation, et lors de la deuxième étape, on a procédé à une optimisation qui affecte la structure du système et plus particulièrement ses entrées (optimisation structurelle), en sélectionnant parmi l'ensemble des indicateurs ceux en mesure de séparer nettement les différentes classes de défauts étudiés. Les indicateurs sélectionnés sont utilisés ensuite comme nouvelles entrées du système de diagnostic spécial lors de cette deuxième étape, On dit alors que la structure du système de diagnostic a été optimisée. La méthode utilisée pour la sélection automatique des indicateurs est le critère de Fischer.

Dans la troisième et dernière étape, et toujours dans le but d'optimiser le système de diagnostic (système à inférence floue (SIF)), et contrairement à l'étape précédente, le changement affecte cette fois les paramètres du système (optimisation paramétrique), en utilisant un système neuro-flou appelé ANFIS(ANFIS : Adaptive Neuro- Fuzzy Inference System). Ce système intégré, qui combine les avantages des deux techniques , la logique

floue(LF) et les réseaux de neurones(RN), ajuste les différents paramètres des fonctions d'appartenance associées aux différentes variables d'entrée et de sortie du système de diagnostic (FIS) par le processus d'apprentissage afin d'obtenir les meilleurs résultats lors du diagnostic des défauts étudiés.

Les différents programmes développés au cours des trois étapes de la méthodologie d'optimisation du système (FIS) ont été testés et validés à l'aide d'un ensemble de signaux vibratoires d'un banc d'essais de roulements en différents modes de fonctionnement (sain et avec défauts). Les résultats obtenus démontrent la capacité et l'efficacité de la méthodologie adoptée pour optimiser le système de diagnostic grâce à une nette amélioration des résultats du diagnostic après le processus d'optimisation par rapport aux résultats avant l'optimisation.

Mots clés:

Diagnostic des défauts mécaniques, "Logique floue", Système à inférence floue(SIF), Optimisation des systèmes à inférence floue, système d'inférence neuro-flou adaptatif(ANFIS), Traitement du signal, Sélection des indicateurs.

Contenu de la Thèse

| | |
|--|-----|
| Résumé..... | i |
| Table des matières..... | iii |
| Liste des figures..... | iii |
| Liste des tableaux..... | iii |
| Liste des abréviations et notations..... | iii |

| | |
|------------------------------------|----------|
| Introduction Générale | 1 |
|------------------------------------|----------|

Chapitre I : Systèmes à Inférence Floue

| | |
|--|-----------|
| 1.1 Introduction..... | 6 |
| 1.2 Théorie des ensembles flous..... | 6 |
| 1.2.1 Ensembles flous..... | 8 |
| 1.2.2 propriétés d'un ensemble flou..... | 9 |
| 1.2.3 Opérations sur les ensembles flous..... | 11 |
| 1.2.4 les α – coupes associées à un ensemble flou..... | 12 |
| 1.3 Diagnostic par les systèmes à base de la logique floue..... | 14 |
| 1.4 Traitement et expertise d'un classificateur flou..... | 15 |
| 1.4.1 Relation floue..... | 15 |
| 1.4.2 Variable linguistique..... | 16 |
| 1.4.3 Modificateur linguistique..... | 16 |
| 1.4.4 Proposition floue..... | 16 |
| 1.4.5 Implication floue..... | 17 |
| 1.5 Structure et fonctionnement d'un système d'inférence flou(SIF)..... | 18 |
| 1.5.1 La Fuzzification | 18 |
| 1.5.2 Le Moteur d'inférence..... | 19 |
| 1.5.3 La défuzzification..... | 21 |
| 1.6 Caractéristiques d'un système à inférence flou..... | 25 |
| 1.6.1 Caractéristiques structurelles | 25 |
| 1.6.2 Caractéristiques paramétriques..... | 26 |
| 1.7 Optimisation d'un système à inférence flou..... | 26 |
| 1.7.1 optimisation paramétrique | 26 |

| | |
|--------------------------------------|-----------|
| 1.7.2 Optimisation structurelle..... | 26 |
| 1.8 Conclusion..... | 27 |

Chapitre2: Etat de L’Art de l’Application de la Logique Floue(LF), les Réseaux de Neurons(RN) et du Neuro-Flou Adaptatif(ANFIS) dans le Diagnostic de Défauts des Machines Tournantes

| | |
|--|-----------|
| 2.1 Introduction..... | 28 |
| 2.2 Les réseaux de neurones artificiels (RNA)..... | 29 |
| 2.2.1 Avantages des RNA..... | 30 |
| 2.2.2 Méthodes d'apprentissage..... | 31 |
| 2.2.3 Types de RNA..... | 32 |
| 2.2.4 Inconvénients des RNA..... | 32 |
| 2.2.5 Applications des RNA dans le diagnostic de défauts des machines tournantes..... | 33 |
| 2.3 La logique floue(LF)..... | 37 |
| 2.3.1 Inconvénients de la logique floue..... | 38 |
| 2.3.2 Applications de la logique floue dans le diagnostic des défauts des machines tournantes..... | 38 |
| 2.4 Hybridation des Techniques d'intelligence artificielle (IA) dans le diagnostic des défauts..... | 42 |
| 2.4.1 Système neuro-flou..... | 42 |
| 2.4.1.1 Application du système neuro-flou dans le diagnostic de défauts..... | 43 |
| 2.4.2 Système d'inférence neuro-flou adaptatif (ANFIS)..... | 44 |
| 2.4.2.1 Architecture d'un système ANFIS | 47 |
| 2.4.2.2 Avantages de l'ANFIS..... | 51 |
| 2.4.2.3 Application du système ANFIS dans le diagnostic de défauts..... | 51 |
| 2.5 Conclusion..... | 53 |

Chapitre3: Démarche Expérimentale

| | |
|--|-----------|
| 3.1 Introduction..... | 55 |
| 3.2 Mise en œuvre d'un système d'inférence flou (SIF)..... | 55 |
| 3.2.1 Étape de fuzzification..... | 56 |
| 3.2.2 Étape d'inférence..... | 56 |
| 3.2.3 Étape de défuzzification..... | 56 |
| 3.3 Déroulement du diagnostic des défauts par un système d'inférence floue (FIS)..... | 57 |

| | |
|--|-----------|
| 3.3.1 Acquisition de signaux..... | 58 |
| 3.3.2 Prétraitement des données..... | 58 |
| 3.3.3 Traitement des données..... | 58 |
| 3.3.4 Construction du vecteur forme..... | 58 |
| 3.3.5 Apprentissage et test du système d'inférence floue(FIS)..... | 58 |
| 3.3.6 Exploitation du système d'inférence floue(FIS)..... | 59 |
| 3.4 Implémentation d'un système d'inférence flou (FIS) sous Matlab..... | 59 |
| 3.4.1 Création du système d'inférence flou (FIS)..... | 59 |
| 3.4.2 Fuzzification des variables d'entrée et de sortie..... | 60 |
| 3.4.3 Édition de règles floues | 61 |
| 3.4.4 Défuzzification | 64 |
| 3.4.5 Test du système d'inférence floue (FIS)..... | 64 |
| 3.5 Implémentation d'un système ANFIS (Adaptive Neuro-Fuzzy Inference Systems) sous Matlab..... | 65 |
| 3.5.1 Implémentation d'un système ANFIS par l'interface graphique (ANFIS editor GUI)..... | 65 |
| 3.5.1.1 Ouverture de l'interface (ANFIS editor GUI)..... | 65 |
| 3.5.1.2 Importation des bases de données..... | 66 |
| 3.5.1.3 Initialisation et génération du système FIS | 67 |
| 3.5.1.4 Affichage de la structure du système ANFIS..... | 69 |
| 3.5.1.5 Apprentissage du système ANFIS..... | 69 |
| 3.5.1.6 Validation du système entraîné FIS..... | 70 |
| 3.6 Description du banc d'essais étudié..... | 71 |
| 3.6.1 Introduction..... | 71 |
| 3.6.2 Signaux de roulements "bearing data center"..... | 72 |
| 3.7 Prétraitement et extraction des indicateurs du vecteur forme..... | 75 |
| 3.8 Classification des défauts par les deux systèmes FIS et ANFIS..... | 75 |
| 3.9 Conclusion..... | 75 |

Chapitre 4 Résultats et Discussion

| | |
|--|-----------|
| 4.1 Introduction | 76 |
| 4.2 Diagnostic des défauts de roulements..... | 77 |
| 4.2.1 Création de la base de données..... | 77 |

| | |
|---|------------|
| 4.2.1.1 Préparation des observations..... | 78 |
| 4. 2.1.2 Constitution du vecteur forme (vecteur d'indicateurs)..... | 84 |
| 4.2.2 Choix des classes | 85 |
| 4.2.3 Classification des défauts de roulements..... | 86 |
| 4.2.3.1 Classification des défauts de roulements avec un système (FIS) initial non optimisé (avec des variables d'entrée choisies arbitrairement)..... | 86 |
| 4.2.3.1.1 Implémentation du système d'inférence flou(SIF) initial sous Matlab..... | 87 |
| a) Implémentation du système (FIS) initial par l'interface graphique de la boite à outils «Fuzzy Logic»..... | 87 |
| a.1) Structure du système d'inférence flou(FIS) initial..... | 87 |
| a.2) Fuzzification des variables d'entrée et de sortie..... | 87 |
| a.3) Génération des règles d'inférence..... | 90 |
| a.4) Défuzzification de la variable de sortie..... | 92 |
| b) Implémentation du système(FIS) initial par les commandes de la boite à outils«Fuzzy logic» | 93 |
| 4.2.3.1.2 Classification de la base de données par le système FIS initial(avant l'optimisation)..... | 94 |
| a) Discussion des résultats de classification..... | 94 |
| a.1) Représentation graphique du résultat de classification..... | 94 |
| a.2) Interprétation du résultat de classification par le système (FIS) initial (avant l'optimisation)..... | 96 |
| 4.2.3.2 Classification des défauts de roulements avec un système (FIS) optimisé structurellement (avec des variables d'entrée sélectionnées)..... | 96 |
| 4.2.3.2.1 Sélection des indicateurs par le critère de Fisher..... | 96 |
| 4.2.3.2.2 Constitution du vecteur forme selon les indicateurs sélectionnés (Vecteur d'indicateurs)..... | 100 |
| 4.2.3.2.3 Choix des classes | 101 |
| 4.2.3.2.4 Implémentation du système d'inférence floue (FIS) optimisé structurellement sous Matlab..... | 101 |
| a) Implémentation du système (FIS) optimisé structurellement par l'interface graphique..... | 101 |
| a.1) Structure du système d'inférence flou(FIS) optimisé structurellement ... | 101 |

| | |
|---|------------|
| a.2) Fuzzification des variables d'entrée et de sortie..... | 102 |
| a.3) Génération des règles d'inférence..... | 105 |
| a.4) Défuzzification de la variable de sortie..... | 107 |
| b) Implémentation du système(FIS) optimisé structurellement par l es commandes de la boîte à outils «Fuzzy logic» | 108 |
| 4.2.3.2.5 Classification de la base de données par le système FIS (optimisé structurellement)..... | 108 |
| a) Discussion du résultat de classification..... | 108 |
| a.1) Représentation graphique du résultat de classification..... | 108 |
| a.2) Interprétation du résultat de classification par le système FIS (optimisé structurellement)..... | 109 |
| 4.2.3.3 Classification des défauts de roulements avec un système (FIS) optimisé paramétriquement (à l'aide d'un système ANFIS)..... | 110 |
| 4.2.3.3.1 Implémentation du système ANFIS (Adaptive Neuro-Fuzzy Inference Systems) sous Matlab..... | 110 |
| a) Structure du système ANFIS | 110 |
| b) Importation des bases de données (apprentissage et test)..... | 111 |
| c) Génération et initialisation du système initial (FIS) | 112 |
| d) Classification des données d'apprentissage et de test à l'aide du système initial SIF (avant l'apprentissage)..... | 114 |
| d.1) Interprétation du résultat de classification du système(FIS) initial (avant optimisation)..... | 115 |
| e) Apprentissage du système ANFIS..... | 115 |
| f) Test du système ANFIS entraîné..... | 117 |
| 4.2.3.3.2 Classification des données d'apprentissage et de test par le système (FIS) optimisé paramétriquement par le système (ANFIS)..... | 118 |
| a) interprétation du résultat de classification par le système (FIS) optimisé paramétriquement par le système (ANFIS)..... | 120 |
| 4.3 Conclusion..... | 120 |
| Conclusion Générale et perspectives..... | 123 |
| Références Bibliographiques | 128 |
| Annexes | |

Liste des Figures

Chapitre 1

| | | |
|-------------------|---|-----------|
| Figure 1.1 | Ensemble flou trapézoïdal..... | 9 |
| Figure 1.2 | Tangente hyperbolique..... | 10 |
| Figure 1.3 | Caractéristiques d'un ensemble flou | 14 |
| Figure 1.4 | Structure d'un système flou..... | 18 |
| Figure 1.5 | Exemple de fonctions d'appartenances..... | 19 |
| Figure 1.6 | Fonctions d'appartenances du rapport cyclique..... | 23 |
| Figure 1.7 | Fonctions d'appartenances du rapport cyclique avec les données en sortie du moteur d'inférence et la fonction d'appartenance résultante..... | 23 |
| Figure 1.8 | Résultat de fuzzification par la méthode de la moyenne des maximas..... | 24 |
| Figure 1.9 | Résultat de fuzzification par la méthode du centre de gravité..... | 25 |

Chapitre 2

| | | |
|-------------------|---|-----------|
| Figure 2.1 | Principe de raisonnement flou (Modèle d'inférence floue de Sugeno)..... | 49 |
| Figure 2.2 | structure des couches ANFIS | 49 |

Chapitre 3

| | | |
|-------------------|--|-----------|
| Figure 3.1 | Étapes de la mise en œuvre d'un système à inférence floue (FIS)..... | 56 |
| Figure 3.2 | Organigramme de mise en place du système d'inférence floue (SIF) pour la classification des défauts | 57 |
| Figure 3.3 | Étapes d'implémentation d'un système d'inférence floue (FIS) sous Matlab..... | 59 |
| Figure 3.4 | Étapes d'implémentation d'un système (ANFIS) sous Matlab..... | 65 |
| Figure 3.5 | Interface graphique pour l'implémentation d'un système ANFIS | 66 |
| Figure 3.6 | Boite de dialogue pour entrer le nom de la base de données..... | 66 |
| Figure 3.7 | Affichage des bases de données..... | 67 |
| Figure 3.8 | Initialisation et génération du système FIS..... | 68 |
| Figure 3.9 | Structure du système ANFIS..... | 69 |

| | |
|--|-----------|
| Figure 3.10 Apprentissage du système ANFIS..... | 70 |
| Figure 3.11 Test du système ANFIS entraîné | 71 |
| Figure 3. 12 Banc d'essai de roulements..... | 72 |
| Figure 3.13 Représentation des signaux temporels pour différents diamètres du défaut sur la bague intérieure..... | 74 |

Chapitre 4

| | |
|--|------------|
| Figure 4. 1 Découpage d'un signal en tranches ou sections..... | 79 |
| Figure 4.2 Division en tranches du signal sans défaut..... | 81 |
| Figure 4.3 Division en tranches du signal avec défaut de diamètre 0,028inch..... | 81 |
| Figures : 4.4 à 4.14 variation des 11 indicateurs en fonction des 112 observations..... | 83 |
| Figures : 4.10 à 4.14 variation des 11 indicateurs en fonction des 112 observations..... | 84 |
| Figure 4.15 Schéma bloc du (SIF)..... | 87 |
| Figure 4.16 Fuzzification de la variable d'entrée « le facteur de crête (F_c)»..... | 88 |
| Figure 4.17 Fuzzification de la variable d'entrée «le facteur de dissymétrie (skewness (K_s))»..... | 89 |
| Figure 4.18 Fuzzification de la variable d'entrée «facteur d'aplatissement (Kurtosis(K_u))»..... | 89 |
| Figure 4.19 Fuzzification de la variable de sortie « état du dispositif (état) »..... | 90 |
| Figure 4.20 Défuzzification de la variable de sortie (état)..... | 93 |
| Figure 4.21 Classification de la base de données. (a) sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée)..... | 95 |
| Figure 4.22 Variation du critère de Fisher pour les 11 paramètres..... | 98 |
| Figure 4.23 Critère de Fisher pour les différents indicateurs en ordre décroissant..... | 98 |
| Figure 4.24 Pourcentage expliqué et cumule des différents indicateurs classés par le critère de Fisher..... | 99 |
| Figure 4.25 Schéma bloc du (SIF)..... | 101 |
| Figure 4.26 Fuzzification de la variable d'entrée «énergie de la bande [4000-5000] (E_{B5})»..... | 102 |
| Figure 4.27 Fuzzification de la variable d'entrée «écart_type (ET)»..... | 103 |
| Figure 4.28 Fuzzification de la variable d'entrée «énergie de la bande [3000-4000HZ] (E_{B4}) »..... | 104 |

| | |
|---|------------|
| Figure 4.29 Fuzzification de la variable de sortie « état du dispositif (état) »..... | 104 |
| Figure 4.30 Défuzzification de la variable de sortie (état)..... | 107 |
| Figure 4.31 Classification de la base de données. (a) sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée)..... | 109 |
| Figure 4.32 Schéma bloc du système ANFIS..... | 111 |
| Figure 4.33 Ensemble d'apprentissage..... | 112 |
| Figure 4.34 Ensemble de test..... | 112 |
| Figure 4.35 choix des paramètres du système initial SIF | 113 |
| Figure 4.36 Fuzzification de la variable d'entrée«EB5»..... | 113 |
| Figure 4.37 Fuzzification de la variable d'entrée«ET» | 113 |
| Figure 4.38 Fuzzification de la variable d'entrée«EB4» | 113 |
| Figure 4.39 Fuzzification de la variable de sortie (état)..... | 113 |
| Figure 4.40 Génération des règles floues..... | 113 |
| Figure 4.41 Structure du système initial SIF..... | 114 |
| Figure 4.42 Classification de la base de données (apprentissage et test). (a) sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée)..... | 115 |
| Figure 4.43 Erreur d'apprentissage du système ANFIS..... | 116 |
| Figure 4.44 Fonctions d'appartenance de la seule sortie avant l'apprentissage (colonne de gauche (a)) et après l'apprentissage (colonne de droite(b)) où il y'a un changement pertinent au niveau des valeurs de paramètres..... | 116 |
| Figure 4.45 Fonctions d'appartenance de chaque entrée avant l'apprentissage (colonne de gauche (a)) et après apprentissage(colonne de droite (b)) où l'axe vertical de chaque sous-figure représente le degré d'appartenance..... | 117 |
| Figure 4.46 Résultats de la classification avec l'approche proposée ANFIS après l'apprentissage, (a) résultat d'apprentissage avec 88 échantillons; (b) résultat de test avec 24 échantillons. | 118 |
| Figure 4.47 Classification de la base de données (apprentissage et test) avec le système FIS optimisé par ANFIS : (a)sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée)..... | 119 |

Liste des Tableaux

Chapitre 2

Tableau 2.1 Applications des réseaux de neurones dans le diagnostic de défauts des machines tournantes.....**33**

Chapitre 3

Tableau 3.1 Caractéristiques des défauts de roulements.....**73**

Tableau 3.2 Caractéristiques des roulements et leurs Fréquences des défauts.....**73**

Chapitre 4

Tableaux 4.1 et 4.2 montrent respectivement la matrice des 11 indicateurs pour les 28 tranches du signal sans défaut et signal avec défaut (diamètre (0.007inch)), les indicateurs de 1 à 11 sont respectivement : ET, F_c , A_{cc} , K_s , K_u , E_{BT} , E_{B1} , E_{B2} , E_{B3} , E_{B4} , E_{B5}(**voir Annexe(4.3)**)

Tableau 4.3 Répartition de la base de données en classes.....**85**

Abréviations et Notations

Abréviations

| | |
|------------------|---|
| FIS | F uzzy I nference S ystem(Système à I nférence F loue(SIF)). |
| ANFIS | A daptive N euro- F uzzy I nference S ystem. |
| IA | I ntelligence A rtificielle |
| RN | R éseau de N eurones |
| LF | L ogique F loue |
| RNA | R éseau de N eurones A rtificiel |
| MLP | M ulti- L ayer P erceptron |
| RBF | R adial B asis F unction |
| LVQ | L earning V ector Q uantization |
| RUL | R emaining U seful L ife |
| SOM | S elf O rganizing M aps |
| BPFNN | B ack P ropagation for F eed F orward N etwork |
| BPMP | B ack P ropagation M ultilayer P erceptrons |
| RBF | R adial B asis F unction |
| RNN | R ecurrent or R ecirculation N eural N etwork |
| AR | A uto- R egressive |
| RMS | R oot M ean S quare |
| DC motor | D irect C urrent m otor |
| DWT | D iscret W avelet T ransform |
| MF | M embership F unctions |
| SFIM | S ugeno F uzzy I nference M odel |
| MSE | M ulti- S cale E ntropy |
| AI | A rtificial I ntelligence |
| ANN | A rtificial N eural N etworks |
| FL | F uzzy L ogic |
| HP | H orse P ower |
| KPPV | K P lus P roche V oisins |
| SVM _s | S upport V ector M achine |
| DA | D iscriminant A nalysis |

| | |
|------|---|
| NF | N euro- F lou |
| MF | M embership f unction |
| SFIM | S ugeno f uzzy I nference M odel |
| SEF | S ous E nsemble F lou |
| LMD | L ocal M ean D ecomposition |
| PF | P roduct F unction |
| EMD | E mpirical M ode D ecomposition |
| EEMD | E nsemble E mpirical M ode D ecomposition |
| CFR | C ritère de F isher R égularisé |
| IDE | I mproved D istance E valuation |
| SD | S tandard D eviation |
| RMS | R oot M ean S quare |

Notations

| | |
|---------------------|---|
| f_a | Fonction d'appartenance |
| Supp(A) | Support de l'ensemble A |
| N(A) | Noyau de l'ensemble A |
| H(A) | Hauteur de l'ensemble |
| α - coupe(A) | α - coupe associée d'un ensemble A |
| K_t | Nombre de tranche par signal |
| N_M | Nombre de points par tranche |
| f_{arbre} | Fréquence de rotation de l'arbre |
| f_{cage} | Fréquence caractéristique de la cage |
| f_e | Fréquence d'échantillonnage |
| Δt | Période d'échantillonnage |
| ET | Ecart_Type |
| F_c | Facteur de Crête |
| A_{cc} | Valeur crête à crête |
| K_s | Facteur de dissymétrie(Skewness) |
| Ku | Facteur d'aplatissement(Kurtosis) |
| J ₄₈ | Algorithme de l'arbre de décision |

PUBLICATIONS

Liste des publications réalisées dans le cadre de cette thèse

PUBLICATIONS

1. **Fateh Harrouche^a**, Ahmed Felkaoui (2014) «Automation of fault diagnosis of bearing by application of fuzzy inference system (FIS) » *Mechanics & Industry Vol.15, 477–485 (2014) AFM, EDP Sciences 2014 DOI: [10.1051/meca/2014059](https://doi.org/10.1051/meca/2014059).*

COMMUNICATIONS INTERNATIONALES

1. **Fateh Harrouche**, Ahmed Felkaoui «comparaison entre deux méthodes d'identification des défauts de roulements ; classification par les systèmes d'inférence flou et les systèmes neuro-inférence flou adaptés» *conférence internationale de mécanique et matériaux ICMM14*, 16-17 novembre 2014, Sétif, Algérie.
2. **Fateh Harrouche**, Ahmed Felkaoui «Automating the Diagnosis of Rotating Machines Defects by Applying Fuzzy Logic(FL)» *Fifth international congress design and modeling of mechanical systems CMSM05*, 25- 27 mars 2013 Djerba Tunisia.
3. **Fateh Harrouche**, Ahmed Felkaoui, Hocine Fenineche « Application de la logique floue(LF) pour le diagnostic de défauts des machines tournantes » *Second International Conference on Industrial Engineering and Manufacturing ICIEM'12*, MAY 06th & 07th 2012 Batna Algeria.
4. Hocine Fenineche, Ahmed Felkaoui, **Fateh Harrouche**, « Application des réseaux de neurones artificiels pour l'automatisation de diagnostic vibratoire d'engrenage » *Second International Conference on Industrial Engineering and Manufacturing ICIEM'12*, MAY 06th & 07th 2012 Batna Algeria.

COMMUNICATIONS NATIONALES

1. **Fateh Harrouche**, Ahmed Felkaoui « Application de la logique floue(LF) pour le Diagnostic de Défauts des machines tournantes » *8^{mes} Journées de Mécanique JM'08*, 10-11 Avril 2012, école militaire polytechnique Bordj El -Bahri algerie.

INTRODUCTION GENERALE

Introduction générale

Contexte de travail

L'industrie actuelle utilise des machines de plus en plus complexes, certaines avec des critères de performance extrêmement exigeants. Tenter de diagnostiquer les défaillances de ces systèmes est souvent une tâche difficile et décourageante pour les opérateurs et les responsables de la maintenance des installations. Des machines en panne peuvent entraîner des pertes économiques et des problèmes de sécurité en raison d'arrêts de production inattendus et soudains. Ces machines doivent être surveillées pendant le processus de production pour améliorer la fiabilité de fonctionnement des machines et réduire les indisponibilités. Par conséquent, mener une surveillance efficace de l'état apporte des avantages significatifs à l'industrie (Altmann, 1999), (Baillie et Mathew, 1996). Sans un diagnostic efficace, il est impossible de faire un pronostic. Une progression naturelle est l'automatisation de ce processus de diagnostic orienté vers le travail en mettant en œuvre des stratégies de diagnostic intelligentes afin que des experts ou des techniciens puissent être soulagés de cette tâche relativement coûteuse.

Le choix d'une méthode de diagnostic automatique dépend essentiellement de la connaissance disponible du procédé étudié. Le diagnostic des défauts est généralement effectué selon les phases suivantes: collecte des données, extraction des caractéristiques, détection et identification du défaut. La détection et l'identification des défauts utilisent généralement des approches à base de l'intelligence artificielle (IA). Une de ces approches est la logique floue. Plusieurs travaux de maintenance des systèmes industriels mécanique ont été réalisés en se basant sur la technique de la logique floue (Mechefske, 1998), (Sugumaran et Ramachandran, 2007), (Boutros et Liang, 2007), (Aliustaoglu et Ertunc, 2008).

Le travail effectué dans cette thèse s'inscrit dans le cadre du diagnostic des défauts mécaniques des machines tournantes faisant appel à des techniques d'intelligence artificielle (logique floue, réseaux de neurones artificiels), Plus précisément, l'objectif de cette étude est

d'optimiser un système de diagnostic de défauts mécaniques créé sur une base inspirée de la logique floue.

La méthodologie adoptée pour cette étude repose sur trois étapes: dans la première étape, un système d'inférence floue(SIF) est créé avec des entrées sélectionnées de manière aléatoire parmi les indicateurs disponibles extraits des signaux vibratoires du système étudié. Dans la deuxième et la troisième étape, on procède à une optimisation de ce dernier. La deuxième étape touchera l'aspect structurel du système, où les entrées du système sont sélectionnées parmi les meilleurs indicateurs capables de séparer nettement les différentes classes de défauts formant la base de données, les indicateurs les plus pertinents sont sélectionnés par une méthode dite « méthode de sélection par le critère de Fisher ». Lors de la troisième étape, nous continuons le processus d'optimisation, mais cette fois on se concentre sur l'aspect paramétrique du système. Dans ce but, on utilise un système intégré neuro-flou. Ce système s'appelle ANFIS(Adaptive Neuro-Fuzzy Inference System) .

La validation pratique pour les différents systèmes de diagnostic élaborés au cours des trois étapes précédentes, est effectuée sur une base de données dérivée des signaux vibratoires d'un banc d'essais de roulements. Les indicateurs formant la base de données peuvent être de nature temporelle ou fréquentielle ([Samanta et al, 2001](#)) ;([Jack et Nandi, 2002](#)); ([Wang et Too, 2002](#)); ([Rafiee et al., 2007](#)); ([Kurek et Osowski, 2010](#)); ([Konar et Chattopadhyay, 2011](#)).

Motivation et objectifs

Les méthodes de classification couramment utilisées pour le diagnostic des défauts mécaniques des machines tournantes se divisent en deux grandes catégories. Selon la procédure de classification des données ([Worden et al, 2011](#)) : on distingue, les méthodes de classification *supervisée* et les méthodes de classification *non supervisée*. Parmi les méthodes de classification à *apprentissage supervisé*, on peut citer : la Logique Floue(LF), les Réseaux de Neurones Artificiels (RNA), les Séparateurs à vaste Marge (SVMs), les arbres de décision, et les K plus proches voisins (Kppv). D'autre part, et pour les méthodes de classification à *apprentissage non supervisé*, on peut citer : Algorithme DBSCAN (Density- Based Spatial Clustering of Applications Noise), Algorithme OPTICS (Ordering Points to Identify the Clustering Structure), et Algorithme K-means(K-moyennes) qui est la plus utilisée en diagnostic de défauts. Ces méthodes nécessitent beaucoup d'expertise pour les mettre en œuvre avec succès ([Gryllias et Antoniadis, 2012](#)). En outre, la plupart de ces méthodes non supervisées connaissent encore des problèmes de stabilité, de convergence, et de robustesse.

Le concept de la logique floue (LF) est basé sur la théorie des ensembles flous présentée par Lotfi A. Zadeh de l'université de Californie, Berkeley, en 1965 (**Zadeh, 1965**). La théorie des ensembles flous porte sur l'imprécision ou l'incertitude et elle fournit une méthode pour l'utilisation de données imprécises avec des concepts mathématiques. La logique floue(LF) offre la possibilité de traiter des incertitudes liées aux processus de maintenance et de planification et est utilisée pour améliorer les performances de détection et de prédiction de défauts dans les systèmes mécaniques. Elle a été largement implémentée dans les applications de diagnostic de défauts en raison de ses avantages en raisonnement approximatif et en mise en œuvre des connaissances linguistiques.

Dans les tâches de surveillance des équipements mécaniques, la théorie de la logique floue a été appliquée dans les cas où les modèles mathématiques précis ne sont pas disponibles ou trop complexes, mais où l'on dispose de connaissances vagues, subjectives et empiriques du problème étudié (**Wang et Lei, 2001**). Généralement les connaissances existantes sont construites comme un ensemble de relations floues ou de règles floues sur lesquelles est basé le système global flou. Ces règles ou relations floues peuvent être construites sur la base d'informations fournies par des experts humains.

La logique floue a plusieurs inconvénients dans son application en tant qu'outil de diagnostic des défauts. L'un des inconvénients est la dépendance élevée vis-à-vis de l'acquisition d'une expérience intuitive d'un opérateur expert et ceci affecte l'objectivité du diagnostic des défauts (**Yang et al....2002**). De plus, les systèmes à base de la logique floue ne possèdent aucune fonction d'auto-apprentissage. cette fonction est une caractéristique importante et qui est très demandée dans les processus de diagnostic des défauts en ligne ou en temps réel (**Gao et Ovaska, 2001**).

Un autre inconvénient des systèmes de diagnostic de défauts basés sur la logique floue se trouve dans son processus de conception, où la construction des règles floues qui représentent le comportement du système modélisé dépend de manière critique des connaissances intuitives et l'expérience acquise auprès d'un expert ou des opérateurs. L'expérience d'un expert est essentielle pour développer la représentation de chaque variable liée aux caractéristiques du problème et aux fonctions d'appartenance floue résultantes. Par conséquent, les règles floues développées ne peuvent pas être garanties comme étant optimales.

La motivation principale de cette thèse est la levée de ces inconvénients, en d'autres termes, l'optimisation de ces systèmes de diagnostic des défauts mécaniques, qui reposent sur la logique floue.

Afin de surmonter ces inconvénients, nous allons toucher deux aspects différents du système: le premier aspect concerne la structure du système (optimisation structurelle), où on s'intéresse dans ce travail à l'optimisation du nombre et du choix des variables d'entrée et de sortie de ce dernier. Quant au deuxième aspect qui concerne les paramètres du système (optimisation paramétriques), il s'agit dans ce type d'optimisation de trouver un jeu de paramètres optimal pour chacune des fonctions d'appartenance associées aux différentes variables d'entrée et de sortie. Pour ce deuxième type d'optimisation, la logique floue(LF) est généralement combinée avec d'autres techniques dans ses applications pratiques au diagnostic des défauts. La technique la plus utilisée est celle des réseaux de neurones artificiels (RNA) qui forment ce qu'on appelle un système adaptatif neuro-flou(ANFIS).

Organisation de la thèse

La thèse est composée, essentiellement, de quatre principaux chapitres :

Dans **le premier chapitre**, Nous avons présenté une synthèse sur la théorie des ensembles flous, ainsi que les outils mathématiques nécessaires à leur manipulation. Le diagnostic des systèmes mécanique complexes constitue un domaine d'application intéressant pour l'application de cette théorie. Dans ce cadre, nous avons présenté et analysé la structure interne d'un système à inférence flou, les caractéristiques ainsi que l'optimisation (paramétrique et structurelle) de ce dernier pour la classification des défauts.

Dans le **deuxième chapitre**, nous décrirons brièvement les techniques d'intelligence artificielle couramment utilisées ainsi que certains travaux réalisés utilisant ces techniques dans le diagnostic des défauts mécaniques des machines tournantes. Les techniques d'intelligence artificielle les plus utilisées sont ; les réseaux de neurones artificiels (RNA), la logique floue(LF), les réseaux neuro-flou(RNF) et le réseau de neurones flou adaptatif (ANFIS : [Adaptive Neuro- Fuzzy Inference System](#)). Les avantages et les inconvénients de ces techniques d'IA dans les applications de diagnostic de défauts ont également été détaillés.

Le troisième chapitre, présente l'essentiel de la démarche expérimentale que nous suivrons dans ce travail, en commençant par les différentes étapes de la mise en œuvre du système d'inférence floue (FIS : [Fuzzy Inference System](#)) ainsi que le système neuro-flou adaptatif (ANFIS : [Adaptive Neuro-Fuzzy Inference System](#)) que nous utiliserons pour l'optimisation paramétrique du système FIS. Ensuite, nous présenterons le banc d'essais de

roulements et les signaux vibratoires prélevés sur ce dernier. Ces signaux sont utilisés ensuite pour tester et valider les différents systèmes de diagnostic conçus dans cette étude.

Le quatrième et dernier chapitre est consacré aux résultats et discussions. La méthodologie choisie pour l'optimisation du système de diagnostic à base de la logique floue repose sur trois étapes, L'application de cette méthodologie commence en première étape par la création d'un système d'inférence floue(FIS) initial avec des variables d'entrée choisies d'une manière arbitraire (au hasard). Par la suite et dans la deuxième étape, on procèdera à une optimisation structurelle des variables d'entrée du système et cela par la sélection des indicateurs les plus pertinents parmi l'ensemble des indicateurs disponible qu'on utilisera comme entrées du nouveau système FIS. Pour la méthode de sélection automatique des indicateurs, on utilisera le «critère de Fisher». Lors de la troisième étape et pour une meilleure configuration du système de diagnostic, on appliquera sur ce dernier une deuxième méthode d'optimisation, mais cette fois ci c'est une optimisation paramétrique qui permettra d'ajuster les différentes valeurs des paramètres des fonctions d'appartenance d'entrée et de sortie. Cette deuxième méthode est une méthode hybride basée sur la logique floue et les réseaux de neurones et qui est appelée la méthode ANFIS.

La validation pratique des différents modèles est effectuée sur les signaux vibratoires d'un banc d'essais de roulements.

CHAPITRE 1
SYSTEMES A INFERENCE
FLOUE

Chapitre 1

Systèmes à inférence floue

1.1 Introduction

L'objectif de ce chapitre est de décrire un peu plus en détail l'approche développée dans les systèmes d'inférences Flous. Pour ce faire, nous présentons les notions de base de la théorie des ensembles flous ainsi que les outils mathématiques nécessaires à leur manipulation. Un des objectifs du chapitre est de montrer l'intérêt que présente la logique floue dans l'identification et le diagnostic des défauts mécaniques dans les machines tournantes, ainsi que la structure interne d'un système d'inférence floue (les trois étapes de mise en œuvre d'un système d'inférence floue : fuzzification, moteur d'inférence et défuzzification).

1.2 Théorie des ensembles flous

Les problèmes de représentation et d'utilisation des connaissances sont au centre d'une discipline scientifique relativement nouvelle et en tout cas controversée, qu'on appelle l'intelligence artificielle. Cette technique a mis l'accent, de façon exclusive, sur le traitement symbolique de la connaissance, par opposition à la modélisation numérique utilisée traditionnellement dans les sciences de l'ingénieur, c'est pour cela et jusqu'à une date récente, l'intelligence artificielle a eu un impact limité sur les applications industrielles. Plus récemment, les problèmes de modélisation numérique ont fait un retour dans les applications de l'intelligence artificielle, avec les réseaux de neurones et la logique floue. Les réseaux de neurones proposent une approche implicite de type "boite noire" de la représentation des connaissances, très analogue à la démarche de l'identification des systèmes en automatique. La logique floue est quant à elle plus conforme à l'intelligence artificielle symbolique, qui met en avant la notion de raisonnement, où les connaissances sont codées explicitement. Elle permet de faire le lien entre modélisation numérique et représentation symbolique, ce qui a permis des développements industriels spectaculaires à partir d'algorithmes de traduction de connaissances symboliques en entités numériques et inversement. La théorie des ensembles

flous a également donné naissance à un traitement original de l'incertitude. Les ensembles flous ont également eu un impact sur les techniques de classification (tel que le cas des problèmes de diagnostic en mécanique) et ont contribué à la proposition de nouvelles approches pour l'aide à la décision.

✓ Historique et domaines d'applications de la logique floue

Les prémises de la logique floue sont apparues en 1940 lorsque des chercheurs américains ont commencé à se pencher sur le concept d'incertitude.

En 1965, Lofti Zadeh introduit la théorie des ensembles flous, (fuzzy set theory), il est alors professeur à l'université californienne de Berkeley et deviendra un automaticien de renommée internationale (Earld, 1997).

En 1973 le professeur Zadeh publie un article qui mentionne pour la première fois le terme de valeurs linguistiques, i.e. dont la valeur est un mot et non une lettre.

En 1974 un chercheur Mamdani établit la première application industrielle de la logique floue : la régulation floue d'une chaudière à vapeur. Il introduit ainsi la commande floue dans la régulation industrielle.

Grâce au chercheur japonais M. Sugeno, la logique floue a été introduite au Japon dès 1985 (Huguenin, 2004), et les sociétés japonaises comprirent l'avantage à la fois technique et commercial de cette logique :

- facilité d'implantation
- solution de problèmes multi-variables complexes
- robustesse vis à vis des incertitudes
- possibilité d'intégration du savoir de l'expert

De nombreux produits grand public sont alors lancés sur le marché japonais avec la mention vendeuse : « fuzzy logic inside ». Comme par exemple dans la conception du lave-linge A ša ğo Day Fuzzy de Matsushita (Février 1990).

Il existe de nombreuses applications dans différents domaines, voici une liste non exhaustive de quelques-unes :

- appareils électroménagers (lave-linge, aspirateur, autocuiseur,...etc.),
- systèmes audio-visuels (appareil photo autofocus, caméscope à stabilisateur d'images,
- photocopieur,...)
- systèmes automobiles embarqués (ABS, suspension, climatisation,...etc.)
- systèmes de transport (train, métro, ascenseur,...),
- systèmes de décision, diagnostic (médical, assurance)

– systèmes de contrôle/commande dans la plupart des domaines industriels de production, transformation, traitement de produits et déchets.

1.2.1 Ensembles flous

Dans la théorie des ensembles classiques, il n'y a que deux situations acceptables pour un élément, appartenir ou ne pas appartenir à un sous-ensemble, et sa fonction caractéristique ne prend que deux valeurs possibles "0" ou "1". Dans le monde réel les classes d'objets rencontrées ne possèdent pas de critères d'appartenance bien définis. Ce constat ne fait que souligner le fossé qui sépare les représentations mentales de la réalité des modèles mathématiques usuels (à base de la logique binaire, de nombres réels, d'équations différentielle, etc.). Le mérite de Zadeh a été de tenter de sortir de cette logique booléenne en introduisant la notion d'appartenance pondérée: permettre des graduations dans l'appartenance d'un élément à un sous-ensemble, c'est-à-dire d'autoriser un élément à appartenir plus moins fortement à ce sous-ensemble, par l'introduction d'une fonction d'appartenance des objets à la classe, prenant des valeurs entre 0 et 1.

Les classes d'objets auxquelles Zadeh fait allusion n'existent qu'au travers de ces représentations mentales et correspondent à des termes vagues du langage naturel, tels que "température élevée", "puissance moyenne", etc. Par exemple, si l'on considère le concept "homme jeune", il est difficile de proposer un seuil en dessous duquel un homme sera considéré comme "jeune". l'idée de Zadeh a été de suggérer qu'au lieu de chercher, à tout prix, un seuil unique pour l'appartenance à l'ensemble des "jeunes" dans un contexte donné, il semblait plus réaliste de considérer deux seuils $s_1 < s_2$, tels que le terme jeune s'applique parfaitement aux âges plus petits que s_1 (par exemple 19 ans), et ne s'applique plus du tout au dessus du s_2 , les âges plus petits que s_1 auront le degré d'appartenance maximale (en général supposé égal à 1) et les âges plus grands que s_2 (par exemple 40 ans) auront un degré d'appartenance minimale (en général égal à 0). Entre s_1 et s_2 , les degrés d'appartenance seront intermédiaires, par convention entre 0 et 1.

➤ Définition

Soit X , un ensemble dénombrable ou non. Un sous-ensemble flou A de X est caractérisé par sa fonction d'appartenance f_A , telle que :

$$f_A : X \rightarrow [0, 1]$$

$$x \rightarrow f_A(x)$$

Où $f_A(x)$ représente le degré d'appartenance avec lequel x appartient à l'ensemble flou A .

➤ **Exemple**

L'intervalle flou (I) couramment utilisé dans l'ensemble des réels (R) est décrit par sa fonction d'appartenance f_I . Le plus simple type pour ce qu'il est convenu d'appeler un «intervalle flou (I)» est une représentation trapézoïdale :

La notation $I = (a, b, \alpha, \beta)$, est souvent utilisée dans les applications informatiques pour représenter une telle fonction, qu'on traduit par :

$$f_I(x) = \begin{cases} 0 & \text{si } x \leq a - \alpha \text{ ou } x \geq b + \beta \\ 1 & \text{si } a \leq x \leq b \\ 1 + \frac{x-a}{\alpha} & \text{si } a - \alpha \leq x \leq a \\ 1 - \frac{x-b}{\beta} & \text{si } b \leq x \leq b + \beta \end{cases} \dots\dots\dots (1.1)$$

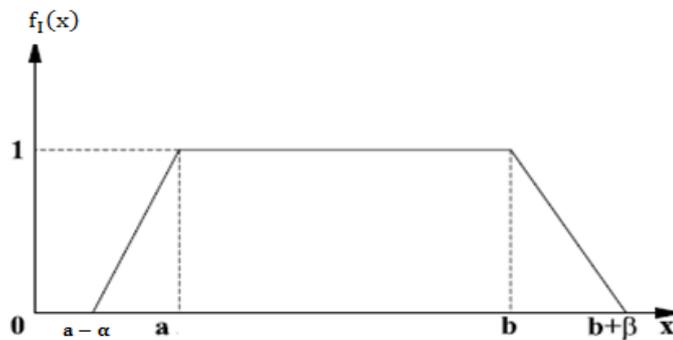


Figure 1.1 Ensemble flou trapézoïdal

1.2.2 Propriétés d'un ensemble flou

Un sous-ensemble flou est complètement défini par sa fonction d'appartenance. A partir d'une telle fonction, un certain nombre de caractéristiques du sous-ensemble flou peuvent être étudiées. Les ensembles flous ont le grand avantage de constituer une représentation mathématique de labels linguistiques largement utilisés dans l'expression de connaissances expertes, qualitatives et manipulées dans un raisonnement approché

Ils apparaissent donc comme un moyen de réaliser l'interface entre l'information numérique (quantitative) et l'information symbolique (linguistique, qualitative). Pour montrer dans quelle mesure un sous-ensemble flou A d'un référentiel X diffère d'un sous ensemble classique de X , nous définissons les caractéristiques essentielles suivantes.

➤ **Définitions**

1. Support d'un sous-ensemble flou

Le support d'un sous-ensemble flou A est caractérisé par l'ensemble des éléments de X, qui ont des degrés d'appartenances supérieur à zero, et que l'on note par $supp(A)$ défini par :

$$supp(A) = \{x \in X / f_A(x) > 0\} \dots\dots\dots (1.2)$$

2. Noyau d'un sous-ensemble flou

Un sous-ensemble flou est normalisé si sa hauteur $h(A) = 1$. Le noyau d'un ensemble flou A de X, noté $N(A)$, est l'ensemble de tous les éléments qui appartiennent totalement ou de façon certaine à A, c'est-à-dire dont le degré d'appartenance à A vaut 1. Formellement on écrit :

$$N(A) = \{x \in X / f_A(x) = 1\} \dots\dots\dots (1.3)$$

3. Hauteur d'un sous-ensemble flou

La hauteur H d'un sous-ensemble flou A est le plus fort degré avec lequel un élément de X appartient à A, la hauteur est définie à l'aide de l'opérateur Sup telle que :

$$H(A) = sup\{f_A(x) / x \in X\} \dots\dots\dots (1.4)$$

➤ **Remarque**

Le sup n'est pas forcément atteint par la fonction d'appartenance f_A . En prenant l'exemple de la fonction tangente hyperbolique ($\forall x \in X, f_A(x) = \tanh(x)$), les caractéristiques du sous-ensemble flou A sont décrits de la façon suivante :

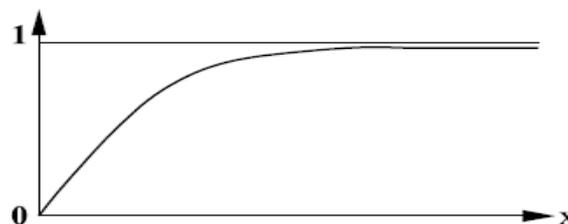


Figure 1.2 Tangente hyperbolique

1. $H(A)=1$, sans qu'il soit atteint ;
2. $N(A)= \emptyset$;
3. $Supp(A)= R$; Le symbole R désigne l'ensemble des nombres réels

4. α -Coupe associée à un ensemble flou

Une partie floue A de X peut aussi être caractérisée par l'ensemble de ses α -coupes.

Une α -coupe d'une partie floue A est le sous-ensemble net (classique) des éléments ayant un degré d'appartenance supérieur ou égal à α .

$$\alpha - \text{coupe}(A) = \{x \in X / \mu_A(x) \geq \alpha\} \dots \dots \dots (1.5)$$

1.2.3 Opérations sur les ensembles flous

Etant donné que le concept de sous-ensemble flou peut être vu comme une généralisation du concept d'ensemble classique (de Cantor) (**Cantor, 1884**), on est conduit à introduire des opérations sur les sous-ensembles flous qui sont équivalentes aux opérations classiques de la théorie des ensembles lorsqu'on a affaire à des fonctions d'appartenance à valeurs 0 ou 1. Zadeh a ainsi généralisé les opérations d'égalité, d'inclusion, de complémentation, d'union, et d'intersection, aux ensembles flous. On présente ici, les opérations les plus couramment utilisés

1. Egalité

On dit que deux ensembles flous A_i et A_j de X sont égaux, si leurs fonctions d'appartenance prennent la même valeur pour tous les éléments x de X .

Formellement $A_i = A_j$ si et seulement si :

$$\forall x \in X, f_{A_i}(x) = f_{A_j}(x) \dots \dots \dots (1.6)$$

2. Inclusion

Soit A_i et A_j deux sous-ensembles flous de X . Si pour n'importe quel élément x de X , x appartient toujours moins à A_i qu'à A_j , alors on dit que A_i est inclus dans A_j ($A_i \subset A_j$).

Formellement, $A_i \subset A_j$ si et seulement si :

$$\forall x \in X, f_{A_i}(x) \leq f_{A_j}(x) \dots \dots \dots (1.7)$$

3. Intersection

L'intersection de deux ensembles flous A_i et A_j de X est l'ensemble flou constitué des éléments de X affectés du plus petit des degrés avec lesquels ils appartiennent à A_i et A_j .

Formellement, $A_i \cap A_j$ est donné par :

$$f_{A_i \cap A_j}(x) = \min(f_{A_i}(x), f_{A_j}(x)), \forall x \in X \dots \dots \dots (1.8)$$

4. Union

L'union de deux ensembles flous A_i et A_j de X est l'ensemble flou constitué des éléments de X affectés du plus grand des degrés avec lesquels ils appartiennent à A_i et A_j .

Formellement, $A_i \cup A_j$ est donné par :

$$f_{A_i \cup A_j}(x) = \max(f_{A_i}(x), f_{A_j}(x)), \forall x \in X \dots \dots \dots (1.9)$$

5. Complémentation

Le complément d'un ensemble flou A_i de X est le sous-ensemble flou noté \bar{A}_i constitué des éléments x lui appartenant d'autant plus qu'ils appartiennent peu à A_i .

Formellement, \bar{A}_i est donné par :

$$f_{\bar{A}_i}(x) = 1 - f_{A_i}(x), \forall x \in X \dots \dots \dots (1.10)$$

1.2.4 les α – coupes associées à un ensemble flou

Il est souvent intéressant de se référer à des sous-ensembles classiques correspondant de façon approximative à des sous-ensembles flous donnés, **afin d'établir des critères de prise de décision.**

La façon la plus simple de réaliser cette approximation est de fixer une limite inférieure, notée α , aux degrés d'appartenance. On construit ainsi le sous-ensemble ordinaire A_α de X associé au sous-ensemble flou A appartenant à $F(X)$ pour le seuil α , en sélectionnant tous les éléments de X qui appartiennent à A avec un degré au moins égal à la valeur du réel α .

➤ Définition

Une α – coupe d'un ensemble flou A ou ensemble de niveau α , noté A_α , est le sous-ensemble (au sens de la théorie des ensembles classiques (ordinaires)) des éléments ayant un degré d'appartenance supérieur ou égal à α .

On dit que α est l' α -coupe de A . formellement :

$$A_\alpha = \{x \in X: f_A(x) \geq \alpha\} \text{ avec } 0 \leq \alpha \leq 1 \dots\dots\dots(1.11)$$

et A_α est un sous-ensemble ordinaire dont la fonction caractéristique est :

$$\chi^{A_\alpha(x)} = \begin{cases} 1 & \text{si } f_A(x) \geq \alpha \\ 0 & \text{si non} \end{cases} \dots\dots\dots(1.12)$$

Si A est un sous-ensemble flou d'un univers X , de fonction d'appartenance f_A , on a :

$$\forall x \in X, f_A(x) = \sup_{\alpha \in]0,1]} \alpha \chi^{A_\alpha(x)} \dots\dots\dots(1.13)$$

(Théorème de décomposition).

D'autre part, on peut définir l'ensemble :

$$A'_\alpha = \{x \in X: f_A(x) > \alpha\} \text{ avec } 0 \leq \alpha \leq 1 \dots\dots\dots(1.14)$$

L'ensemble A'_α est appelé α – coupe fort ou α – coupe stricte ;

➤ Propriétés des α -coupes

Lorsqu'on construit une α - coupe A_α du sous-ensemble flou A , on peut dire que α représente le seuil d'appartenance, relativement à la définition de A , plus on est exigeant sur la notion d'appartenance, plus on augmente ce seuil.

$\forall \alpha_1$ et $\alpha_2 \in [0,1]$, on a les propriétés suivantes:

1. si $\alpha_1 \geq \alpha_2 \Rightarrow A_{\alpha_1} \subseteq A_{\alpha_2}$
2. $(A \cap B)_\alpha = A_\alpha \cap B_\alpha$
3. $(A \cup B)_\alpha = A_\alpha \cup B_\alpha$
4. $B \subseteq A \Rightarrow B_\alpha \subseteq A_\alpha$
5. Pour le réel α appartenant à l'ensemble $\{0,1\}$, deux cas sont possibles :
 - Pour $\alpha = 0 \Rightarrow A_\alpha = X$
 - Pour $\alpha = 1 \Rightarrow A_\alpha = N(A)$

➤ Exemple

La Figure 1.3 montre les différentes caractéristiques d'un ensemble flou ainsi que la α -coupe .

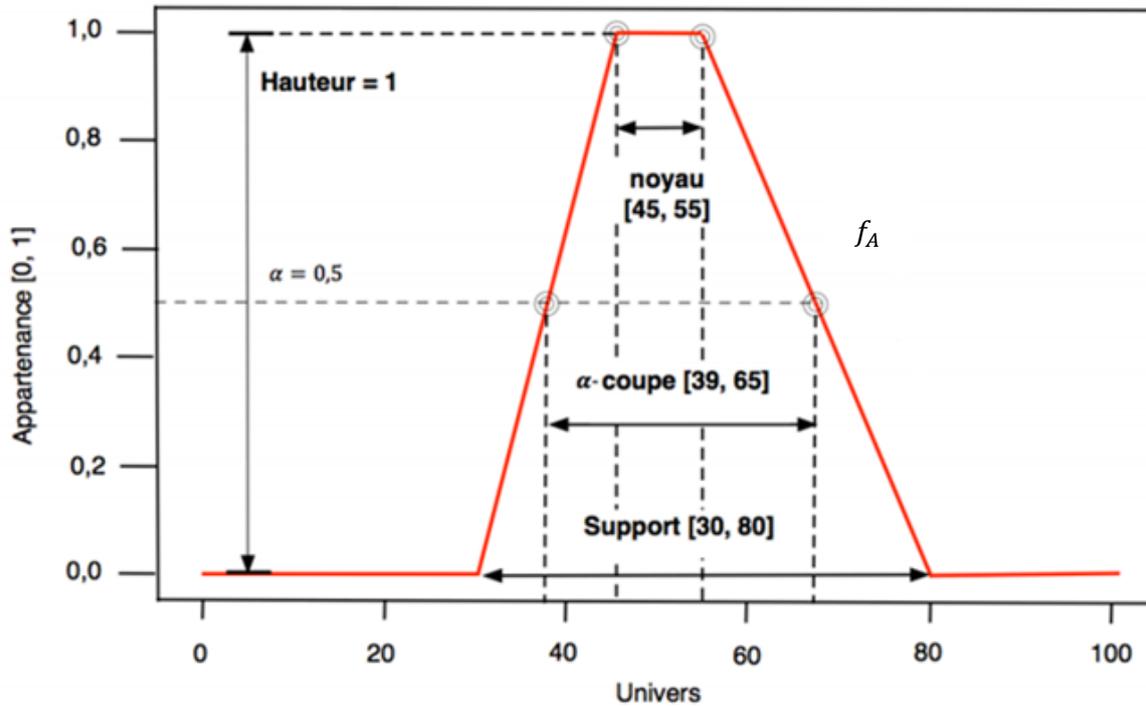


Figure 1.3 Caractéristiques d'un ensemble flou

1.3 Diagnostic par les systèmes à base de la logique floue

Un processus dynamique de nature mécanique est une entité susceptible d'évoluer en fonction du temps. On suppose que le comportement dynamique de ce processus n'est pas satisfaisant vis-à-vis d'un certain nombre de critères de bon fonctionnement. L'objectif du diagnostic est d'agir sur ce processus afin d'atteindre un comportement désiré (un bon fonctionnement). La résolution de ce problème consiste à calculer l'évolution des variables du modèle proposé au comportement du processus puis à faire un diagnostic pour voir la où se situe le défaut dans le processus dynamique (de nature mécanique), et par la suite le corriger.

Les objectifs de recherche en diagnostic mécanique visent à la maîtrise de la surveillance de systèmes de plus en plus complexes. De tels systèmes sont en majorité non-linéaires, multivariables et présentent des couplages internes. Ces systèmes sont souvent caractérisés par des modèles difficiles à établir, dont la résolution nécessite l'emploi d'outils mathématiques complexes. Le succès de cette résolution dépend du degré de la complexité du problème (dimension, nature des non-linéarités, couplage entre variables, caractère stochastique, ..., etc.).

En mécanique, l'absence totale de modèle de connaissance ou d'un modèle précis du système rend dans certains cas l'application des méthodes classiques de diagnostic difficile, voir

impossible pour la synthèse d'une solution performante. Les techniques basées sur de la logique floue constituent dans ce cas une alternative intéressante. En effet, ces techniques permettent d'une part, de formaliser le traitement de l'inconsistance et de l'incertitude sur le modèle du système et d'autre part, d'établir une interopérabilité du comportement dynamique du système par un ensemble de règles linguistiques, traduites en langage simple et interprétable. Une des applications les plus représentatives de la logique floue est celle du diagnostic des défauts mécaniques par des systèmes à base de l'inférence floue (SIF).

Ces travaux de thèse s'inscrivent dans ce cadre et plus précisément dans l'optimisation d'un système de diagnostic des défauts de roulements basé sur la logique floue (inférence floue).

1.4 Traitement et expertise d'un classificateur flou

Selon Zadeh la modélisation floue fournit des moyens approximatifs mais efficaces pour décrire le comportement des systèmes qui sont trop complexes ou trop mal définis pour admettre l'utilisation d'une analyse mathématique précise (Zadeh, 1975), (Zadeh, 1983). Les classificateurs flous utilisent, en général, une expertise proche du langage humain sous forme de règles, dont la forme générale, pour un classificateur flou à deux entrées et une sortie, est décrite par l'expression suivante : **si** x_1 est A_1 et x_2 est A_2 **alors** y est B . Cette dernière s'appelle règle floue et le membre de gauche est appelé prémisse de la règle, tandis que l'expression (Y est B) représente sa conclusion. Les variables x_1 et x_2 et y représentent les variables physiques caractéristiques du processus à diagnostiquer, et du problème de diagnostic associé. A_1 , A_2 et B sont des valeurs linguistiques, qui sont représentées par des ensembles flous.

Dans ce qui suit, nous donnerons quelques définitions utiles pour la compréhension du mécanisme de raisonnement flou.

1.4.1 Relation floue

Une relation floue R entre X et Y est définie par un sous-ensemble flou du produit cartésien $X \times Y$. en particulier, si X et Y sont finis, cette relation peut être décrite à l'aide de la matrice $M(R)$ des valeurs de sa fonction d'appartenance f_R .

1.4.2 Variable linguistique

L'expression « variable linguistique » introduite par (Zadeh, 1971), propose que les valeurs de cette variable ne soient pas numériques mais symboliques comme les mots du langage courant (naturel) ou dans un langage artificiel. Une **variable linguistique** est donc une

variable prenant ses valeurs dans un ensemble de mots symboliques (sous-ensemble flous) définissant certaines catégories d'un ensemble de référence. Elle est représentée par le triplet suivant : (V, X, T_v) où V représente la variable du problème, X l'univers de discours et $T_v = \{A_1, A_2, A_3, \dots, A_n, \dots\}$ un ensemble fini ou non de sous-ensemble flous de X , qui caractérise la variable V .

1.4.3 Modificateur linguistique

Un modificateur linguistique est un opérateur ψ qui permet, à partir de toute caractérisation floue A_i de V , l'obtention d'une nouvelle caractérisation floue $\psi(A)$. Si la fonction d'appartenance de A est f_A , celle de $\psi(A)$ est donnée par la formulation suivante $f_{\psi(A)}(x) = t_\psi f_A(x)$. Cette dernière est obtenue à partir d'une transformation mathématique t_ψ attachée à ψ . Pour un ensemble M de modificateurs quelconques, on note $M(T_v)$ l'ensemble des caractérisations floues, engendré par M à partir de T_v .

Exemple

Soient T_v et M deux ensembles, définis de la manière suivante :

$$T_v = \{\text{petit}, \text{moyen}, \text{grand}\}$$

$$M = \{\text{plutôt}, \text{non}\}$$

L'ensemble des caractérisations floues, engendré par M à partir de T_v est décrit par l'ensemble suivant :

$$M(T_v) = \{\text{plutôt petit}, \text{plutôt moyen}, \text{plutôt grand}, \text{non grand}, \text{non petit}, \text{non moyen}\}$$

1.4.4 Proposition floue

Le traitement et la représentation des connaissances symboliques non numériques, passent par le conditionnement des données sous forme de variables linguistiques. On se limite à des descriptions de la forme suivante ; la taille est moyenne, la température est élevée, etc. formellement, on considère un ensemble L de variables linguistiques et un ensemble de modificateurs M .

Définition 1

Une proposition floue élémentaire est définie à partir d'une variable linguistique (V, X, T_V) de L et par la qualification $(V \text{ est } A_i)$. La valeur de vérité de cette proposition, est donnée par sa fonction d'appartenance f_{A_i} .

Définition 2

Une proposition floue générale est obtenue par la composition de propositions floues élémentaires. L'utilisation de la proposition floue générale la plus simple, s'exprime par la conjonction de propositions floues élémentaires sous la forme suivantes :

$$(V \text{ est } A_i) \text{ et } (W \text{ est } A_j)$$

Où V et W sont des variables définies sur les référentiels X et Y .

La proposition floue générale est associée au produit cartésien $A_i \times A_j$ qui caractérise la valeur conjointe (V, W) sur l'ensemble $X \times Y$.

La valeur de vérité de cette proposition est définie par le nombre réel :

$$\min(f_{A_i}(x), f_{A_j}(y)), \forall x \in X \text{ et } \forall y \in Y$$

1.4.5 Implication floue

Une implication floue est une proposition floue qui possède la forme suivante :

$$\text{si } (V \text{ est } A_i) \text{ alors } (W \text{ est } A_j)$$

D'une manière générale, les propositions floues sont construites à l'aide de différents opérateurs, comme la disjonction, la conjonction ou bien l'implication portant sur des propositions floues quelconques. La particularité d'une proposition floue est qu'elle utilise une implication floue entre deux propositions floues ayant la forme suivante :

$$\text{si } (V \text{ est } A_i) \text{ et } (W \text{ est } A_j) \text{ alors } (U \text{ est } A_k)$$

On parle également dans ce cas de **règle floue**.

1.5 Structure et fonctionnement d'un système d'inférence flou (SIF)

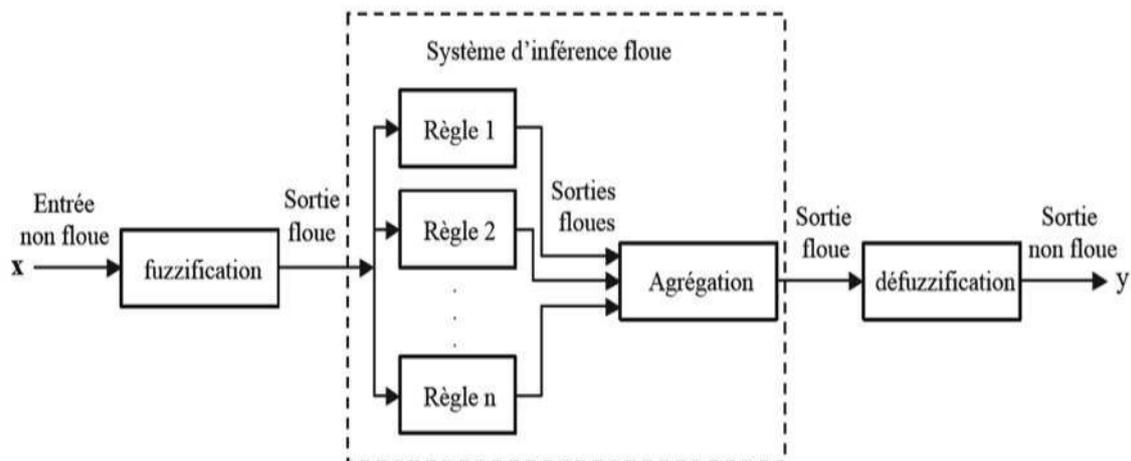


Figure 1.4 Structure d'un système flou

La structure conventionnelle d'un système d'inférence flou (SIF) est présentée par la Figure 1.4. Elle est composée de trois étapes distinctes dont les définitions sont données ci-dessous.

1.5.1 La Fuzzification

L'étape de fuzzification a pour but de transformer une donnée numérique en variable linguistique. Pour cela, le concepteur du système flou doit créer des fonctions d'appartenance. Une fonction d'appartenance est une fonction qui permet de définir le degré d'appartenance d'une donnée numérique à une variable linguistique.

Prenons par exemple une température en degré Celsius provenant d'un capteur. On veut transformer cette donnée numérique en variable linguistique. Bien évidemment, on peut utiliser plusieurs variables linguistiques pour caractériser un seul type de données. Ici, nous choisirons trois variables linguistiques pour qualifier la température : chaud, froid et tempéré. Pour cela, il faut créer une fonction d'appartenance pour chaque variable. Comme ces fonctions d'appartenance qualifient un même type de données, on peut les représenter sur le même graphique (voir Figure 1.5).

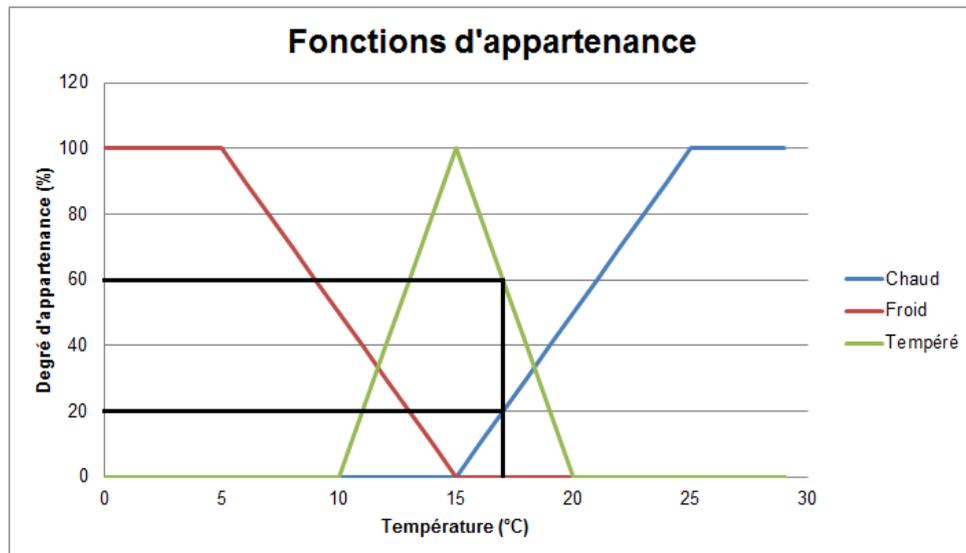


Figure 1.5 exemple de fonctions d'appartenances

De la Figure 1.5, Si le capteur nous renvoie 17 °C, après fuzzification, la température sera chaude à 20%, tempérée à 60% et froide à 0%.

1.5.2 Le Moteur d'inférence

Maintenant que l'on possède des variables linguistiques, on va pouvoir les passer dans le moteur d'inférence. Ici, chaque règle du moteur d'inférence est écrite par le concepteur du système flou en fonction de connaissance qu'il possède. La première chose à faire pour cette seconde étape est donc de lister toutes les règles que l'on connaît et qui s'applique au système. Une règle doit être sous la forme «*Si condition, alors conclusion*». Par exemple, «*SI la vitesse est grande ET la distance au feu est courte ALORS freine fort* », est une règle d'inférence valide.

Le problème dans les règles d'inférences, c'est de savoir ce que les opérateurs logiques signifient. En effet, les opérateurs de la logique classique (ET, OU) ne sont plus valables en logique floue. Il faut donc les redéfinir nous-mêmes.

a) L'opérateur ET

L'opérateur ET en logique floue correspond à l'intersection de deux ensembles flous. Il existe plusieurs définitions de l'opérateur ET en logique floue. Parmi les plus utilisés, on a :

- L'opérateur de minimalité : $a \text{ ET } b = \min(a, b)$
- L'opérateur produit : $a \text{ ET } b = a.b$

Si on sait que la vitesse est grande à 80% ($a=0,8$) et que la distance au feu est courte à 20% ($b=0,2$) alors, je vais freiner fort à 20%. $\min(80\%, 20\%)=20\%$

$$a.b=0,8 \times 0,2=0,16 =16\%$$

Par exemple, reprenons la règle d'inférence ci-dessus (*Si la vitesse est grande ET la distance au feu est courte ALORS freine fort*) en utilisant l'opérateur de minimalité Si on sait que la vitesse est grande à 80% ($a=0,8$) et que la distance au feu est courte à 20% ($b=0,2$) alors, je vais freiner fort à 20% ($\min(80\%, 20\%)=20\%$). Tandis que si j'utilise l'opérateur produit, je freinerai fort à 16% ($a.b=0,8 \times 0,2=0,16 =16\%$).

Il existe une troisième interprétation de l'opérateur ET logique qui se nomme le ET flou. Cet opérateur est un mixe entre l'opérateur de minimalité et la moyenne arithmétique. Le ET flou se définit comme il suit:

$$a \text{ ET } b = \gamma \cdot \min(a,b) + (1-\gamma) \cdot (a+b)/2$$

Le ET flou possède un paramètre γ qui est compris entre 0 et 1 et qui doit être fixé par le concepteur du système flou. Si γ vaut 1, on se retrouve avec l'opérateur de minimalité tel que défini plus haut.

Si on reprend l'exemple ci-dessus, mais en utilisant le ET flou avec un γ à 0.5 : $80\% \text{ ET } 20\% = 0.5 \min(80\%, 20\%) + 0.5 (80\% + 20\%) / 2 = 35\%$. Je devrais donc freiner fort à 35%.

b) L'opérateur OU

L'opérateur OU en logique floue correspond à l'union de deux ensembles flous. Comme pour le ET logique, il existe plusieurs définitions du OU logique dans le cadre de la logique floue :

- L'opérateur de maximalité: $a \text{ OU } b = \max(a, b)$
- L'opérateur produit : $a \text{ OU } b = 1 - (1-a) \cdot (1-b)$

Si nous modifions la règle d'inférence ci-dessus pour avoir un OU dans les conditions : *Si la vitesse est grande OU la distance au feu est courte ALORS freine fort* Dans ce cas, si on utilise l'opérateur de maximalité, je freinerais fort à 80%, et avec l'opérateur produit, je freinerais fort à 84%.

De même que pour le ET, il existe un troisième opérateur assez connu pour le OU logique : le OU flou. Cet opérateur est défini comme une combinaison entre l'opérateur maximum et la moyenne arithmétique.

$$a \text{ OU } b = \gamma \cdot \max(a, b) + (1-\gamma) \cdot (a+b) / 2$$

Si l'on choisit le OU flou avec un gamma de 0.5, il faudra donc freiner fort à 65%.

c) L'opérateur NON

L'opérateur NON en logique floue correspond à l'ensemble complémentaire et est défini simplement : $\text{NON } a = 1 - a$.

Bien sûr, le choix des opérateurs revient entièrement au concepteur du système flou. Celui-ci peut même décider de créer ses propres opérateurs qui remplaceront les opérateurs logiques moyennant le respect de certaines propriétés algébriques (associativité, commutativité, monotonie et, respectivement, 0 comme élément absorbant et 1 comme élément neutre pour un opérateur ET et 1 comme élément absorbant et 0 comme élément neutre pour un opérateur OU).

Une fois que l'on a dressé une liste de règles d'inférences, que l'on a choisi les opérateurs logiques que l'on souhaiterait utiliser, il suffit d'appliquer chaque règle aux variables linguistiques calculés dans l'étape de fuzzification. Les résultats de ces règles pourront directement aller à l'étape finale de défuzzification.

1.5.3 La défuzzification

La dernière étape pour avoir un système flou opérationnel s'appelle la défuzzification. Lors de la seconde étape, on a généré un tas de commandes sous la forme de variables linguistiques (une commande par règle). Le but de la défuzzification est de fusionner ces commandes et de transformer les paramètres résultants en donnée numérique.

L'étape de défuzzification se déroule en deux temps :

D'abord, il faut fusionner les variables linguistiques communes à l'aide d'un opérateur de la logique floue choisi par le concepteur du système. Si on a plusieurs règles d'inférences qui génèrent plusieurs valeurs de la même variable linguistique, on peut choisir un opérateur pour combiner les valeurs de la variable. Cet opérateur sera dans la grande majorité des cas, l'opérateur OU logique utilisant l'opérateur de maximalité. Par exemple, si on a trois règles qui génèrent la variable linguistique *accélération fortement* à 20%, 25% et 35%. Il en résultera que la variable *accélération fortement* aura pour valeur finale 35%.

Dans un second temps, nous pouvons réellement entamer la partie délicate de la défuzzification. On a une série de variables linguistiques qui caractérisent une seule et même donnée. **Par exemple**, on peut avoir trois variables linguistiques : *accélération forte* à 35%, *accélération moyenne* à 80% et *accélération faible* à 0% qui qualifie l'accélération. Ces

variables linguistiques possèdent chacune une fonction d'appartenance. Défuzzifier la donnée d'accélération revient donc à trouver la meilleure valeur quantitative en fonction des fonctions d'appartenances des variables linguistiques.

Il existe plusieurs méthodes pour défuzzifier. Parmi les plus utilisés, on peut citer la méthode de **la moyenne des maximas** et la méthode du **centre de gravité**

La méthode de la moyenne des maximas revient à prendre l'abscisse correspondant à la moyenne des abscisses ayant pour ordonnée la valeur maximale des fonctions d'appartenance. Dans le cas précédent, le maximum de la fonction d'appartenance, c'est 80% pour la variable linguistique *accélération moyenne*. Il faut donc faire la moyenne des abscisses pour lesquels la fonction d'appartenance *accélération moyenne* est supérieure ou égale à 80%.

D'un point de vue formel, la méthode de la moyenne des maximas s'exprime de cette façon:

$$valeur = \frac{\int_S x \cdot dx}{\int_S dx} \text{ avec } S = \{x, \mu(x) = \sup(\mu(x))\}$$

La seconde méthode est la méthode du centre de gravité Elle consiste à prendre l'abscisse correspondant au centre de gravité de la fonction d'appartenance. Formellement, on l'exprime

$$valeur = \frac{\int_S \mu(x) \cdot x \cdot dx}{\int_S \mu(x) \cdot dx} \text{ comme : avec } S, \text{ le domaine de la fonction d'appartenance.}$$

Pour illustrer ces deux méthodes de défuzzification, **prenons un exemple**. On souhaite régler le rapport cyclique d'un moteur à l'aide d'un système flou. On fournit au système des données brutes et un ensemble de règles d'inférence. Le système calcul automatiquement les sorties sous forme de variables linguistiques. On suppose que les variables linguistiques de sorties sont *rapport cyclique faible* à 0%, *rapport cyclique moyen* à 80% et *rapport cyclique fort* à 30%.

On considère que l'on a choisi les fonctions d'appartenances suivantes pour chacune des trois variables linguistiques (voire Figure 1.6) :

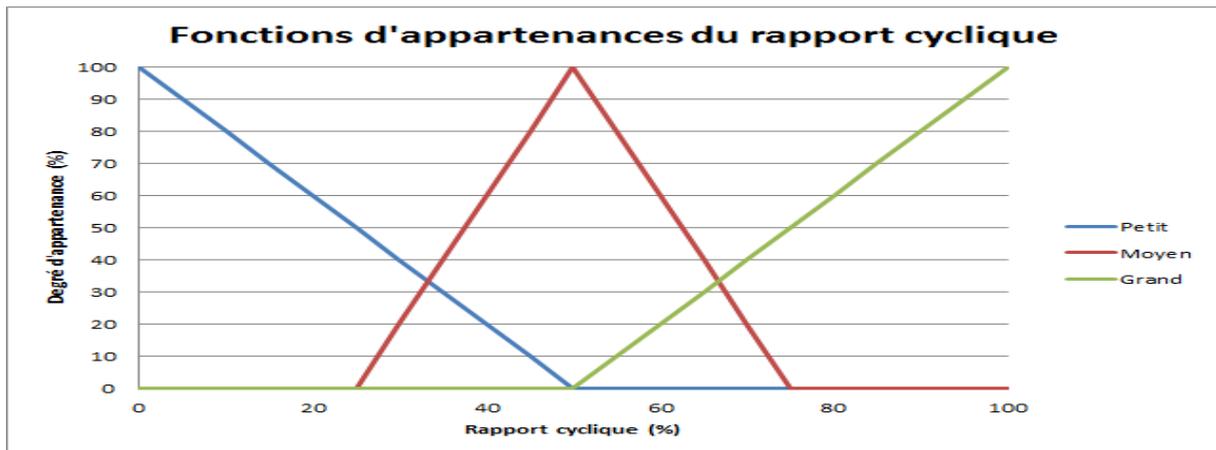


Figure 1.6 Fonctions d'appartenances du rapport cyclique

Avec les données en sortie du moteur d'inférence, on est capable de générer la nouvelle fonction d'appartenance (voire Figure 1.7) :

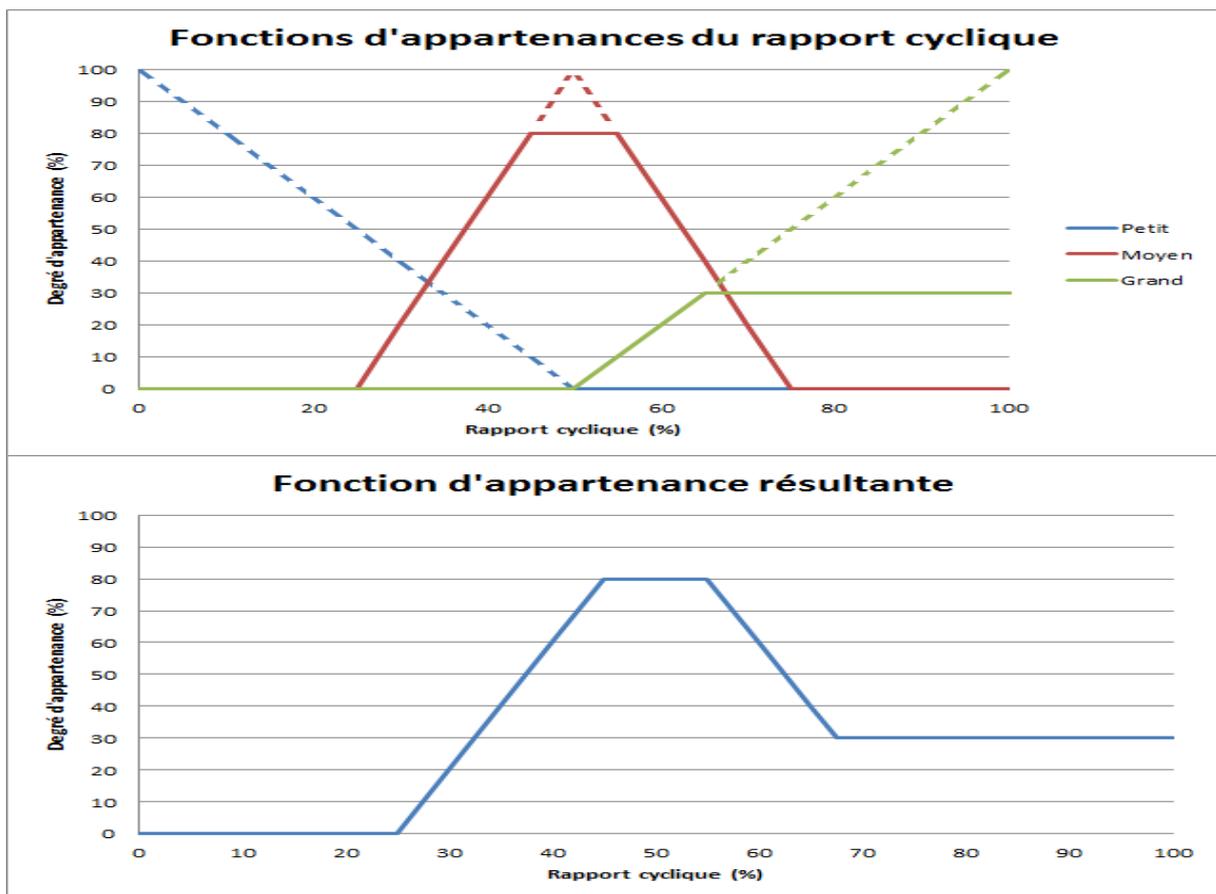


Figure 1.7 Fonctions d'appartenances du rapport cyclique avec les données en sortie du moteur d'inférence et la fonction d'appartenance résultante.

Ainsi, pour défuzzifier, si l'on applique la première méthode de la moyenne des maximas, on remarque que la fonction d'appartenance résultante est maximale pour un rapport cyclique compris entre 40 et 60%. Il suffit de faire la moyenne sur l'intervalle [40,60] pour trouver le rapport cyclique résultant. Ici, le résultat de la défuzzification vaudra 50% (voire Figure 1.8).

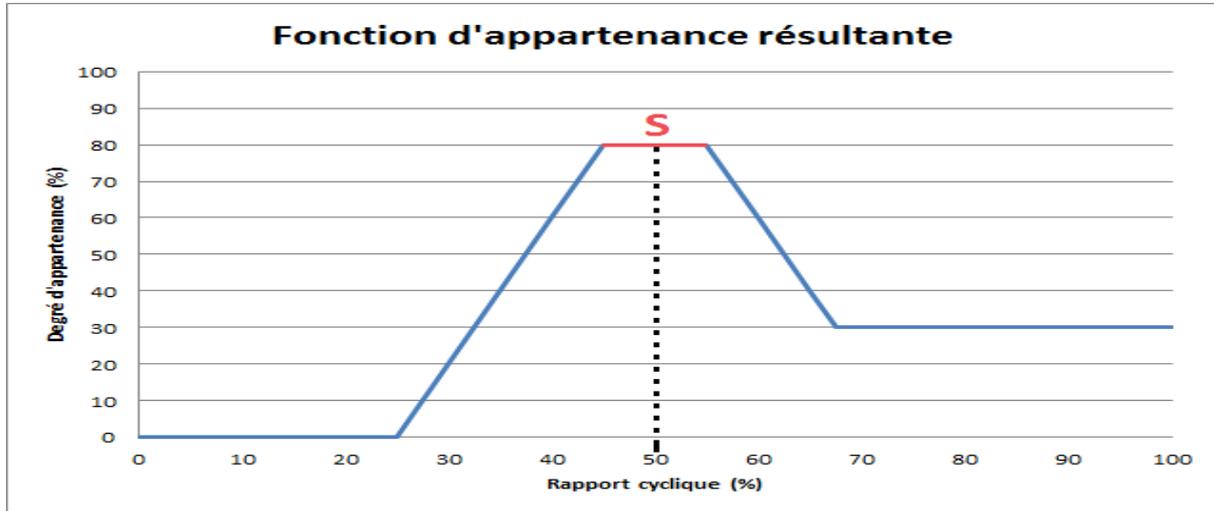


Figure 1.8 Résultat de fuzzification par la méthode de la moyenne des maximas

La seconde méthode du centre de gravité donne des résultats bien meilleurs et est largement utilisée dans les systèmes flous. Néanmoins, elle possède l'inconvénient d'être très coûteuse. En effet, pour appliquer cette méthode de défuzzification, il faut calculer le centre de gravité de la surface sous la fonction d'appartenance et de prendre l'abscisse de ce centre de gravité. Pour cela, il faut décomposer la fonction d'appartenance en petits morceaux et intégrer sur chacun des morceaux.

$$\begin{cases} \mu(x) = (x - 25\%) \cdot 4 & x \in [25\%, 45\%] \\ \mu(x) = 80\% & x \in [45\%, 55\%] \\ \mu(x) = 80\% - (x - 55\%) \cdot 4 & x \in [55\%, 67.5\%] \\ \mu(x) = 30\% & x \in [67.5\%, 100\%] \end{cases}$$

Il suffit maintenant d'appliquer l'intégrale du centre de gravité :

$$\begin{aligned} \text{valeur} &= \frac{\int_{25\%}^{45\%} \mu(x) \cdot x \cdot dx + \int_{45\%}^{55\%} \mu(x) \cdot x \cdot dx + \int_{55\%}^{67.5\%} \mu(x) \cdot x \cdot dx + \int_{67.5\%}^{100\%} \mu(x) \cdot x \cdot dx}{\int_{25\%}^{45\%} \mu(x) \cdot dx + \int_{45\%}^{55\%} \mu(x) \cdot dx + \int_{55\%}^{67.5\%} \mu(x) \cdot dx + \int_{67.5\%}^{100\%} \mu(x) \cdot dx} \\ &= 58\% \end{aligned}$$

La valeur du rapport cyclique calculée lors de la défuzzification par la méthode du centre de gravité vaut donc 58%, ce qui est une valeur plus réaliste (voire Figure 1.9).

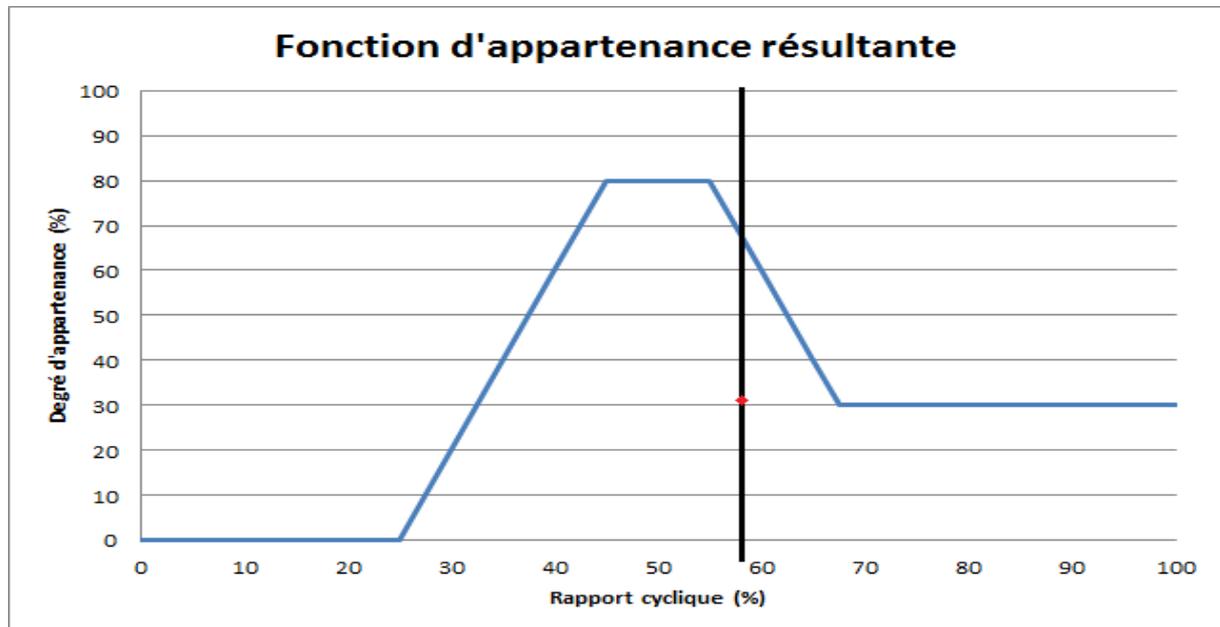


Figure 1.9 Résultat de fuzzification par la méthode du centre de gravité

La défuzzification est une partie délicate à implémenter dans un système flou et elle constitue alors une prise de décision. En effet, elle consomme généralement pas mal de ressource informatique pour pouvoir transformer les variables linguistiques en données numériques, car dans cette partie, on manipule des fonctions. Ceci peut être un point critique dans certains systèmes embarqués et le choix de la méthode de défuzzification y est donc crucial.

1.6 Caractéristiques d'un système à inférence flou

Les systèmes à inférence floue sont caractérisés par le type utilisé (Mamdani, Takagi- Sugene, etc.), par la méthode de partition du domaine de définition ou de validité pour les différentes variables utilisées et aussi par un ensemble de caractéristiques structurelles et paramétriques. Ces caractéristiques sont définies comme suit :

1.6.1 Caractéristiques structurelles

Ces caractéristiques spécifient les composants du système. Le changement de ces éléments affecte la structure du système. Les plus importantes de ces caractéristiques sont:

- 1- Le nombre de variables utilisées pour la génération de la base de règles

- 2- Le nombre de termes linguistiques associés à chaque variable du système
- 3- Le type de fonction d'appartenance utilisé pour caractériser les termes linguistiques (gaussiennes, triangulaires, trapézoïdales, etc.)
- 4- La taille de la base de règles (nombre de règles générées)
- 5- La méthode utilisée pour la défuzzification lors de la mise en œuvre du système à inférence flou.

1.6.2 Caractéristiques paramétriques

Ces caractéristiques spécifient les paramètres du système flou qui représentent l'aspect purement numérique du système, on distingue :

- 1- Les paramètres des fonctions d'appartenances associées aux différentes variables d'entrée (moyenne, écart-type, variance, base, etc.)
- 2- Les paramètres des fonctions d'appartenance associées à la variable de sortie

La mise en œuvre d'un système à inférence flou, dépend du processus qui conduit à la formalisation, sous forme de règles, des relations apprises à partir d'un ensemble de données d'entrées/sorties du système. L'induction des règles utilisée comporte deux étapes de génération de règles, suivie d'une phase d'optimisation du système.

1.7 Optimisation d'un système à inférence flou (Glorennec, 2000)

L'optimisation des systèmes à inférence flou se fait à deux niveaux, le premier niveau concerne la structure du système (optimisation structurelle), le second niveau concerne les paramètres du système (optimisation paramétrique)

1.7.1 Optimisation paramétrique : il s'agit de trouver un jeu de paramètres optimal pour chacune des fonctions d'appartenance associées aux différentes variables d'entrée et de sortie. Au cours de ces dernières années on assiste à l'émergence de nouvelles approches basées sur des techniques numériques, telles que les approches hybrides neuro-flou par l'utilisation des algorithmes d'apprentissage supervisé, par renforcement, ou encore celles qui utilisent les algorithmes génétiques

1.7.2 optimisation structurelle : elle se déroule en deux étapes :

- la sélection des variables d'entrée les plus pertinentes : Il existe de nombreuses méthodes pour la sélection des variables pertinentes, parmi lesquelles on peut citer

la classification des différentes variables selon leurs valeurs **d'indice de pouvoir discriminant (Hong et Chen, 1999)**, la classification selon la valeur de l'indice de variation d'entropie pour chaque variable **(Lei et al., 2008) , (Zhang et al., 2010)**, la classification des variables selon la **valeur du critère de Fisher** pour chaque variable **(Kudo et al., 2000) , (Dubuisson, 1990) , (Casimir, 2003) , (Casimir et al., 2006), (FELKAOUI, 2007)**

- La réduction de la base de règles (déterminer le nombre optimal de règles, le nombre de fonctions pour chaque variable, les opérateurs de conjonction, de disjonction, et d'implication, la technique de défuzzification). Dans plusieurs travaux de recherches **(Saravanan et al, 2009), (Dawu et al, 2009), (Sugumaran et al, 2007)**, les tailles des bases de données disponibles sur les systèmes étudiés sont importantes en volume et par conséquent, la génération automatique de systèmes d'inférence floue nécessite une optimisation structurelle.

1.8 Conclusion

Nous avons présenté dans ce chapitre une synthèse sur la théorie des ensembles flous, ainsi que les outils mathématiques nécessaires à leur manipulation. Nous avons ainsi montré l'intérêt de cette théorie pour formaliser le traitement de l'inconsistance et de l'incertitude sur le modèle d'un système et pour en interpréter le comportement dynamique par un ensemble de règles linguistiques, traduites en langage simple et interprétable. Le diagnostic des systèmes mécanique complexes constitue un domaine d'application intéressant pour l'application de cette théorie. Dans ce cadre, nous avons présenté et analysé la structure interne d'un système à inférence flou, les caractéristiques ainsi que l'optimisation (paramétrique et structurelle) de ce dernier. Notre objectif est de proposer une méthodologie pour la génération et l'optimisation d'un système d'inférence flou pour le diagnostic des défauts de roulements. Dans le chapitre suivant, nous présenterons un état de l'art de l'application de la logique floue et du neuro-flou dans le diagnostic des défauts mécaniques des machines tournantes.

CHAPITRE 2

***ETAT DE L'ART DE L'APPLICATION
DE LA LOGIQUE FLOUE (LF), LES RESEAUX
DE NEURONES(RN) ET DU NEURO-
FLOU ADAPTATIF (ANFIS) DANS LE
DIAGNOSTIC DES MACHINES TOURNANTES***

Chapitre 2

Etat d'Art de l'Application de la Logique Floue(LF), les Réseaux de Neurons(RN) et du Neuro-Flou Adaptatif(ANFIS) dans le Diagnostic de Défauts des Machines Tournantes

2.1 Introduction

La nécessité de mettre en œuvre des systèmes intelligents dans le diagnostic des défaillances est devenue une exigence essentielle dans la mesure où les processus de maintenance industrielle deviennent de plus en plus complexes. Cette complexité est le résultat de développements sophistiqués dans les machines; et les processus industriels impliqués. La quantité de travail exigée actuellement, pour fournir un diagnostic de défauts rapide et fiable, a dépassé les capacités des opérateurs humains à gérer ces processus en temps opportun.

En outre, en raison de la nature compliquée des machines modernes, il existe également un risque accru de défaillance nécessitant des techniques efficaces de résolution des problèmes. Le traitement de défaillance des machines est un facteur important dans l'adaptation avec ; la demande croissante de produits de haute qualité production et réduction du coût, la disponibilité des machines en temps opportun, la fiabilité des équipements de production et le plus important, des exigences de sécurité accrues dans tous les éléments de la chaîne de production.

Ces facteurs sont devenus motivants pour l'utilisation de modèles de diagnostic intelligents dans l'identification des défauts dans les machines.

L'identification de l'état final est une autre tâche dans le diagnostic de défauts des machines tournantes. Le diagnostic est principalement effectué en utilisant des techniques d'intelligence artificielle (IA). Les applications des méthodes d'IA sont principalement caractérisées par l'implémentation des systèmes à base de la logique floue ou de réseaux de neurones.

En général, les éléments constitutifs d'un système intelligent de diagnostic des défauts sont les suivants: section de collecte de données, section d'extraction des indicateurs et la section de détection et d'identification des défauts. La section de détection et d'identification des défauts

peut utiliser les systèmes suivants: réseaux de neurones artificiels (RNA), la logique floue (LF), Systèmes experts et techniques d'intelligence artificielle hybride.

Les sections suivantes décrivent brièvement les techniques d'IA couramment utilisées ainsi que certains travaux réalisés utilisant ces techniques dans le diagnostic des défauts mécaniques des machines tournantes.

2.2 Les réseaux de neurones artificiels (RNA)

Les réseaux de neurones artificiels (RNA) et ses variantes peuvent être l'une des techniques d'intelligence artificielle largement utilisées dans le diagnostic des défauts (**Haykin, 1999**).

Les réseaux de neurones artificiels sont un composant principal de la neuro-informatique (**Rao et al, 2012**). Ils ont été explorés pour la première fois par Rosenblatt en 1959 (**Widrow et Lehr, 1990**), (**Widrow et Lehr, 1960**).

Le réseau de neurones artificiel se compose de groupes de neurones artificiels interconnectés. Ces neurones utilisent un modèle mathématique ou informatique pour le traitement de l'information. Le RNA est un système adaptatif qui change sa structure en fonction de l'information qui circule à travers le réseau (**Haykin, 1999**).

Le réseau de neurones artificiel apprend les échantillons à partir d'un ensemble de données d'apprentissage qui représente la relation entre les entrées et les sorties en utilisant une méthode d'apprentissage (**Rao et al, 2012**). Un exemple de ceci est vu dans un algorithme d'apprentissage de type rétro-propagation, introduit par Werbos en 1974 (**Widrow et Lehr, 1990**).

Le réseau de neurones artificiel comprend une couche de nœuds d'entrée, une ou plusieurs couches de nœuds cachés et une couche de sortie. Chaque couche se compose des éléments de traitement appelés «nœuds» ou «Neurones». L'interaction entre eux est réalisée en utilisant des liaisons (connexions) pondérées numériquement (**Tsoukalas et Uhrig, 1997**).

Le système de diagnostic intelligent qui utilise les réseaux de neurones artificiels est généralement basé sur l'utilisation d'un système de traitement de données qui comprend un nombre d'éléments (nœuds) de traitement simples et fortement interconnectés. Ces éléments fortement interconnectés sont disposés dans une structure qui imite (simuler) le cortex cérébral dans le cerveau humain. Le système est utilisé dans le diagnostic intelligent en fournissant des entrées connues afin d'obtenir une sortie qui représente les caractéristiques de défaut de la machine.

Une fonction inconnue peut être définie via une évaluation et un ajustement itératifs des valeurs des poids de pondération utilisés par le RNA. Dans ce contexte, le RNA peut être utilisé dans la modélisation des problèmes complexes non linéaires et qui peuvent être mis en œuvre pour rapprocher les problèmes pratiques dans les cas de diagnostic de défauts. Dans les applications pratiques de diagnostic, le RNA est entraîné pour reconnaître les échantillons extraits à partir des signaux enregistrés ou acquis. Pendant le processus d'apprentissage, le RNA est chargé d'imiter (de simuler) la relation entrée-sortie des données ou des informations. C'est-à-dire que, le réseau de neurone artificiel(RNA) est entraîné pour mémoriser les relations qui lui permettront d'être utilisé ultérieurement comme classificateur où il est conçu pour classer les formes d'entrée (données / informations) et relier les entrées aux sorties connues.

2.2.1 Avantages des RNA

Le RNA présente plusieurs avantages, ce qui le rend approprié pour une mise en œuvre dans une large gamme d'applications.

Premièrement, le RNA ne dépend pas de principes antérieurs, de données statistiques ou de modèles. Par conséquent, il peut être utilisé directement dans des tâches de modélisation analytiquement difficile et qui ne peuvent pas être résolues en utilisant des méthodes ou des approches conventionnelles. En effet, l'utilisation des méthodes conventionnelles n'est pas pratique car les tâches de modélisation ont des caractéristiques non linéaires, elles sont d'ordre supérieur, sont dynamique et les relations des données entrée-sortie n'ont pas de modèle analytique.

Deuxièmement, le RNA apporte une amélioration aux capacités de tolérance avec les défauts dans les systèmes de diagnostic de défauts grâce à ses caractéristiques adaptatives dans lesquelles il dispose de capacités d'auto-ajustement qui compensent les changements dans les données au fil du temps.

Grâce à ces avantages, les RNA convient parfaitement aux applications dans le diagnostic et l'identification de défauts et qui impliqueraient autrement des tâches de modélisation et des méthodes d'analyse classiques peu pratiques.

En outre, la construction et les méthodes de stockage et la manipulation de l'information au sein des RNA les rendent appropriés à l'apprentissage continu. Cette caractéristique d'apprentissage peut être utilisé pour reconnaître les formes (modes) même si l'information (observation) qui fait la forme est bruité ou incomplète. Ceci est également connu comme la

caractéristique des (RNA) dans l'appariement de données dans des espaces de grandes dimensions et dans laquelle les (RNA) utilise l'efficacité d'interpolation et la capacité d'extrapolation et cela à base de données apprises (d'apprentissage). L'établissement d'une fonction complexe de régression entre un ensemble de données d'entrée et de sortie peut être obtenu en utilisant le processus d'apprentissage des RNA.

Pour des fins de diagnostic de défauts, un RNA est appliqué en utilisant des données d'entrée appropriées, afin d'apprendre les données et de produire des informations sur le défaut représentant les caractéristiques des données.

La capacité de mapper les relations (entrée-sortie) par le processus d'apprentissage en utilisant les données présentées a fait du RNA l'une des méthodes les plus largement utilisées et préférées dans le diagnostic intelligent des défauts des systèmes de machines. Les caractéristiques d'apprentissage sont à la base de larges applications du RNA dans le diagnostic des défauts. Les paramètres sont alors extraits à partir des signaux vibratoires acquis sur des machines en présence de défaut, connu et introduits dans un RNA pour l'apprentissage. Comme résultat d'apprentissage des caractéristiques du système, le RNA peut déterminer la plupart des modes de fonctionnement normaux où anormaux intégrés dans les données. Les modes anormaux peuvent être utilisés comme des signatures associées à des états (modes) de défauts. La capacité d'un RNA à mapper les données (entrée – sortie) par son processus d'apprentissage lui permet d'être utilisé comme un système de reconnaissance de formes en ligne utile dans le processus de diagnostic de défauts.

2.2.2 Méthodes d'apprentissage

Dans l'application du RNA, il existe deux méthodes d'apprentissage ou d'entraînement. La première est l'apprentissage supervisé et la deuxième est l'apprentissage non supervisé (Peng, et al., 2010). L'apprentissage supervisé est le où des exemples complets de données ou de paires de données (entrée-sortie) sont présentés au RNA au cours de sa session d'apprentissage (Haykin, 1999). Les données présentées contiennent la cible ou la sortie connue correspondante (Carpenter et al, 1992). L'apprentissage non supervisé est celui où il n'y a pas de valeurs de sortie (cible) et la tâche d'apprentissage vise à obtenir les caractéristiques du processus qui a généré les données (Kohonen, 1997), (Haykin, 1999).

Les étapes d'apprentissage et de test sont deux procédures importantes dans l'application d'un système RNA. L'étape d'apprentissage joue un rôle très important puisque c'est là que le RNA gagne ses capacités utiles. L'étape de test est le processus par lequel les entrées avec des

caractéristiques connues sont introduites dans le RNA pour tester sa performance et pour effectuer des calculs de pondération de réseau afin de produire des valeurs pour le dernier stade neuronal. Ces valeurs sont ensuite comparées à la sortie ciblée afin de vérifier la pertinence de la conception du RNA.

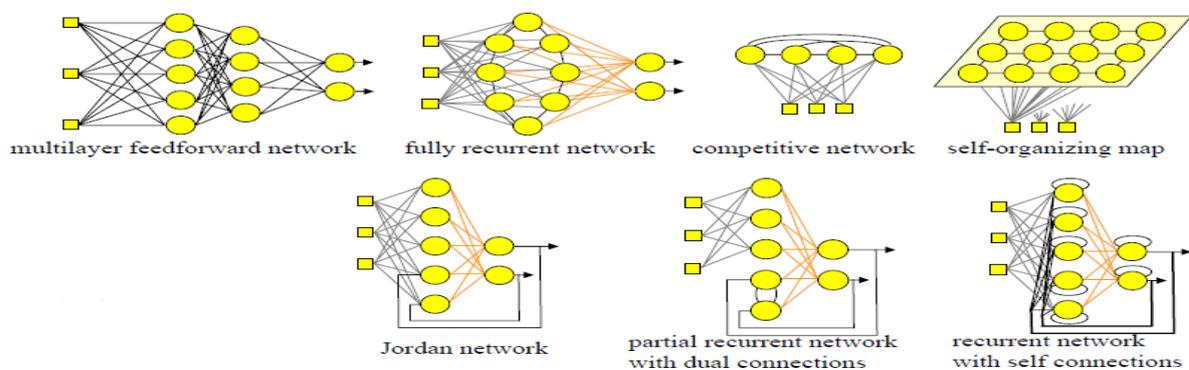
2.2.3 Types de RNA

Il y a plusieurs types de RNA (Jang et al, 1997). Nous mentionnons parmi eux:

- Back Propagation for Feed Forward Network (BPFFN)
- Multi Layer Perceptrons (MLP)
- Back Propagation Multilayer Perceptrons (BPMP)
- Self Organized Map (SOM)
- Radial Basis Function (RBF)
- Recurrent or Recirculation Neural Network (RNN)
- Learning Vector Quantization(LVQ).

Tous ces types de RNA ont été mis en application dans des diverses applications de diagnostic de défauts.

Quelques architectures :



2.2.4 Inconvénients des RNA

Bien que les RNA présentent des avantages, ils présentent également des inconvénients. Premièrement, un RNA appliqué comme un outil de diagnostic agit en tant que boîte noire, ce qui signifie qu'il est difficile de déduire des explications physiques pour sa sortie. Deuxièmement, le processus d'apprentissage devient plus compliqué et prend du temps une fois que la taille du RNA augmente. Troisièmement, il y a un problème dans la détermination du nombre de couches cachées à utiliser ainsi que le nombre de nœuds dans chaque couche (Brotherton, et al, 2000).

Enfin, il manque au RNA une fonctionnalité sémantique nécessaire pour traiter le caractère imprécis et vague des informations caractéristiques du raisonnement humain. Dans ce contexte, la forme de base des réseaux de neurones ne peut pas être employée pour adapter à l'utilisation de la connaissance experte acquise à partir de l'observation et de l'étude d'un processus physique (Liu et Shi, 2001).

2.2.5 Applications des RNA dans le diagnostic de défauts des machines tournantes

Les réseaux de neurones artificiels sont actuellement les techniques les plus couramment trouvées basées sur les données dans la littérature sur le diagnostic de défauts (Heng et al. 2009) et ils ont été largement appliqués dans le diagnostic des défauts des machines tournantes. Des exemples d'applications des RNA dans le diagnostic de défauts sont répertoriés dans le Tableau 2.1, dans lequel les applications aux groupes de machines courants sont rapportés.

Tableau 2.1 Applications des réseaux de neurones dans le diagnostic de défauts des machines tournantes

| Type de réseau de neurone | moteurs | pompes | roulements | turbines | Engrenages et boîtes de vitesses | Arbres(axes) |
|---------------------------|--|---------------------------|---|-------------------------|--|---|
| BPFFN | (Mahamad et Hiyama, 2011), (Li et al. 2004) | (Ilott et Griffiths,1997) | (Li and Wu, 1989), (Liu et Mengel, 1992), (Samanta et Al-Balushi, 2003), (Gebraeel et al. 2004), (Sreejith, et al. 2008) (Wang et al. 2010), (Prieto et al. 2013) | | (Yang et al. 2008), (Wu et Chan, 2009) | (McCormick et Nandi, 1997), (Kuo et al. 2002) |
| RNN | | | (Yam et al. 2001), (Malhi et al. 2011) | (Mohammadi et al. 2011) | | |
| RBF | (Selaimia, 2006), (Onel et Benbouzid, 2008) | (Lu et al. 2011) | | (Rong et al. 2009) | (Li et al. 2009) | |
| MLP | (Paya et al., 1997) | | (Meesad et Yen, 2000), (Senguler et Al, 2010) | | (Meesad et Yen, 2000) | (Meesad et Yen, 2000) |

| | | | | | | |
|-------------|--|--|--|---|---------------------|-----------------------|
| Kohonen SOM | (Kowalski et Kowalska, 2003), (Bay et Bayir, 2005) | | | | | |
| SOM | (Yang et al. 2004), (Premrudee-arn et al. 2002) | | (Zhang et Ganesan, 1997), (Zhong et al. 2005), | (Hu et al. 2003), (Wu et al. 2002), (Donat et al. 2008) | (Liao et al. 2005) | |
| LVQ | | | (Zarei et al. 2008) | | (Abu-Mahfouz, 2005) | (Meesad Et Yen, 2000) |

Plusieurs applications sélectionnées répertoriées dans le Tableau 2.1 sont explicitées ci dessous.

Les systèmes à base des RNA ont été utilisés pour soutenir la classification de l'analyse des défauts. Une application très tôt des RNA dans le domaine de l'analyse des défauts de roulements peut être trouvée dans le travail de (Li et Wu, 1989). Dans cette application, un réseau de type perceptron a été utilisé pour analyser les données expérimentales des roulements à billes. Les résultats ont montré que le réseau a identifié les défauts criés sur la bague extérieure des roulements avec un pourcentage d'erreur plus petit que celui de l'un des méthodes conventionnelles. Il a été signalé que la technique proposée a atteint un taux meilleur de 14% que celui des méthodes conventionnelles.

Dans une recherche menée par (Liu et Mengel, 1992), il a été montré que le réseau de type perceptron était capable de distinguer entre six cas différents de défauts de roulement à billes. La détection de défauts a utilisé les variations de l'amplitude de pic dans le domaine fréquentiel, le pic RMS et des paramètres du spectre de puissance comme des données d'apprentissage pour le réseau de type perceptron.

(Baillie et Mathew, 1994) ont diagnostiqués défauts des éléments roulants de roulements à l'aide des réseaux de neurones et pour ce la un système de diagnostic des défauts de roulements a été développé. Le signal de vibration entrant a été présenté à chaque modèle de réseau neuronal dans le système et le modèle de réseau qui a le mieux rapproché le signal a été choisi d'indiquer un type de défaut. Le système a été entraîné pour diagnostiquer les cas de défaut tels que le déséquilibre, les défauts de la bague extérieure, les défauts de la bague intérieure et le cas sans défauts (normale). Le système de diagnostic basé sur les réseaux de neurones a été testé et il a été démontré que le système atteignait une précision de 95% dans toutes les bases de données de test.

(McCormick et Nandi, 1997) ont utilisé Les deux types de réseaux de neurones Perceptron multicouches (Multi-Layer Perceptron (MLP)) et la Fonction de base radiale (Radial Basis Function (RBF)) pour classer les différents états de défauts des machines tournantes. Dans les tâches de classification, les résultats ont été montrés que des taux de réussite similaires ont été atteints par les deux réseaux MLP et RBF. Plus de détails montrent que le réseau RBF a besoin d'un temps d'apprentissage beaucoup plus court que le temps nécessaire pour le réseau MLP. Cependant, le réseau MLP a atteint un temps de fonctionnement plus rapide et a utilisé moins de neurones.

(Meesad et Yen, 2000) ont appliqué le réseau MLP et la quantification du vecteur d'apprentissage (Learning Vector Quantization (LVQ)) tant que classificateurs dans le diagnostic des défauts d'engrenages, de roulements, et d'arbres. Les deux réseaux ont été mise en oeuvre avec succès. Un apprentissage hors ligne et une alimentation itérative de données étaient nécessaires pour réussir le processus de classification des défauts.

(Zhang et Ganesan, 1997) ont appliqué une estimation de tendance multi-variable du développement de défauts pour prédire la durée de vie utile restante (Remaining Useful Life (RUL)) d'un système de roulement en utilisant un réseau neuronal auto-organisateur. La surveillance d'état du système a été effectuée par des mesures de vibration en ligne et la quantification du défaut a été formulée dans une analyse de tendance multi-variable. Les réseaux de neurones auto-organiseurs ont été utilisés pour effectuer la tendance multi-variable du développement de défaut. Il a été constaté que la précision de l'algorithme de prédiction proposé était la même que celle de l'un des algorithmes SOM(Self Organized Map).

(Yam et al, 2001) ont étudié la tendance en prédiction de l'état d'une machine en utilisant un système de réseau de neurone récurrent (recurrent neural-network(RNN)).

(Kowalski et Kowalska, 2003) ont démontré une application de réseaux de neurones pour le diagnostic des défauts du moteur à induction. Dans cette recherche deux types de RNA étaient proposés l'un est un réseau de type perceptron multicouche(MLP) et l'autre de type réseau de Kohonen auto-organisé(SOM). Les résultats des tests expérimentaux ont montré que les réseaux de neurones pouvaient être utilisés efficacement pour la reconnaissance des défauts

du stator, rotor, roulement et des défauts asymétriques d'alimentation par des mesures appropriées et l'interprétation de l'analyse FFT des spectres de vibration courants.

(Gebraeel et al, 2004) ont mené une étude sur le pronostic des roulements de poussée afin d'essayer de déterminer la prévision du temps réel de défaillance du roulement. L'étude visait à développer des modèles basés sur les réseaux de neurones pour prédire la défaillance des roulements. Une installation expérimentale a été développée pour effectuer des essais de roulements où des informations sur les vibrations ont été collectées à partir d'un certain nombre de roulements qui ont fonctionné jusqu'à la défaillance. Ces informations ont été ensuite utilisées pour l'apprentissage des modèles de réseaux de neurones pour la prévision des temps de fonctionnement des roulements. Les données de vibration provenant d'un ensemble de validation de roulements ont été ensuite appliquées à ces modèles de réseau. Les prédictions résultantes ont été ensuite utilisées pour estimer le temps de défaillance des roulements.

(Sreejith et al, 2008) ont proposé une application des réseaux de neurones pour le diagnostic automatisé des défauts localisés dans les roulements. Le kurtosis et la classification des log-vraisemblances (log-likelihood classification) **(Goumas et al., 2001) et (Abbasion et al., 2007)** extraites à partir des signaux vibratoires du domaine temporel, ont été utilisés comme paramètres d'entrée pour le réseau de neurone. Les résultats ont montré que le réseau de neurone entraîné était capable de classer les différents états de défauts avec un taux de précision de 100%.

(Wang et al, 2010) ont utilisé la méthode autorégressive (autoregressive(AR)) combinée au réseau de neurones avec rétropropagation (backpropagation neural network (BPNN)) pour le diagnostic des défauts des machines tournantes. Une nouvelle méthode de diagnostic des défauts a été étudiée en utilisant les différences dans les paramètres AR avec le réseau BPNN. Les résultats du diagnostic obtenus ont été comparés avec trois méthodes : BPNN avec des paramètres AR, BPNN avec les différences de coefficients et AR et BPNN avec des distances de coefficients AR. Il a été constaté que les résultats du diagnostic obtenus en utilisant BPNN avec des différences de coefficient AR étaient supérieurs aux résultats des deux autres méthodes.

(Prieto et al, 2013) ont combiné des paramètres statistiques temporels et les réseaux de neurones pour la détection des défauts de roulements dans les moteurs électriques. Des paramètres statistiques temporels tels que ; l'erreur quadratique moyenne (Root Mean Square(RMS)), l'écart type (Standard Deviation(SD)), la variance et le facteur de crête ont été calculés à partir des signaux vibratoires acquis. Les valeurs de l'analyse discriminante (the discriminant analysis (DA)) ont été utilisées pour la sélection des paramètres. Les tâches finales de classification ont été effectuées en utilisant la structure hiérarchique du réseau de neurone et l'efficacité de la méthode a été vérifiée par les résultats expérimentaux obtenus à partir de différents états de fonctionnement. La méthode proposée a atteint un taux de bonne classification de 95% de l'ensemble de test global.

(Khajavi et al, 2016) ont utilisé un système à base des réseaux de neurones de type perceptron multicouches (MLP) pour prédire l'usure des outils lors du usinage. À cette fin, une série d'expériences a été menée à l'aide d'une fraiseuse sur une pièce CK45. L'usure de l'outil a été mesurée à l'aide d'un microscope optique. Pour améliorer la précision et la fiabilité du système de surveillance, l'état d'usure de l'outil a été classé en cinq classes: absence d'usure, usure légère, usure normale, usure sévère et outil cassé. Des expériences ont été menées avec les états d'usure d'outils susmentionnés et sous différentes conditions d'usinage et les données ont été extraites. Une augmentation de l'amplitude du courant a été observée lorsque l'usure de l'outil augmentait. De plus, les effets de paramètres tels que l'usure de l'outil, l'avance et la profondeur de coupe sur la consommation de courant du moteur ont été analysés. Compte tenu de la complexité de la classification de l'état d'usure, un réseau de neurones de type perceptron multicouches a été utilisé. Les valeurs quadratiques moyennes du courant moteur, de l'avance, de la profondeur de coupe et du régime de rotation de l'outil ont été choisies comme entrées et la quantité d'usure du flanc comme sortie du MLP. Les résultats ont montré une bonne performance du système conçu pour la surveillance d'état d'usure de l'outil.

2.3 La logique floue(LF)

Le concept de la logique floue (LF) est basé sur la théorie des ensembles flous présentée par Lotfi A. Zadeh de l'université de Californie, Berkeley, en 1965 (Zadeh, 1965). La théorie des ensembles flous est au sujet de l'imprécision ou l'incertitude et elle fournit une méthode pour l'utilisation de données imprécises dans des concepts mathématiques

2.3.1 Inconvénients de la logique floue

La logique floue a plusieurs inconvénients dans son application en tant qu'outil de diagnostic des défauts. L'un des inconvénients est la dépendance élevée vis-à-vis de l'acquisition d'une expérience intuitive d'un opérateur expert et ceci affecte l'objectivité du diagnostic des défauts (Yang et al...2002). De plus, les systèmes à base de la logique floue manquent des possibilités de n'importe quelle fonction d'auto-apprentissage, cette fonction est une caractéristique importante et qui est très demandée dans les processus de diagnostic des défauts en ligne ou en temps réel et qui nécessitent des résultats de haute précision (Gao et Ovaska, 2001).

Un autre inconvénient des systèmes de diagnostic de défauts basés sur la logique floue se trouve dans son processus de conception, où la construction des règles floues qui représentent le comportement du système modélisé dépend de manière critique des connaissances intuitives et l'expérience acquise auprès d'un expert ou des opérateurs. L'expérience d'un expert est essentielle pour développer la représentation de chaque variable liée aux caractéristiques du problème et aux fonctions d'appartenance floue résultantes. Par conséquent, les règles floues développées ne peuvent pas être garanties comme étant optimales.

Afin de surmonter ses inconvénients, la logique floue(LF) est généralement combinée avec d'autres techniques dans ses applications pratiques au diagnostic des défauts. La technique la plus largement utilisée en combinaison avec la (LF) est celle des réseaux de neurones artificiels (RNA) et qui forment ce qu'on appelle un système neuro-flou.

2.3.2 Applications de la logique floue dans le diagnostic des défauts des machines tournantes

La logique floue(LF) offre la possibilité de traiter des incertitudes liées aux processus de maintenance et de planification et elle a été utilisée pour améliorer les performances de détection et de prédiction de défauts dans les systèmes mécaniques. Elle a été largement implémentée dans les applications de diagnostic de défauts en raison de ses avantages en raisonnement approximatif et en mise en œuvre des connaissances linguistiques.

Dans les tâches de surveillance des équipements mécaniques, la théorie de la logique floue a été appliquée dans les cas où les modèles mathématiques précis n'étaient pas disponibles ou

trop complexes, mais où il y avait encore des connaissances vagues, subjectives et empiriques sur le problème étudié (Wang et Lei, 2001). Généralement les connaissances existantes sont construites comme un ensemble de relations floues ou des règles floues sur lesquelles est basé le système global flou. Ces règles ou relations floues peuvent être construites sur la base d'informations fournies par des experts humains.

Dans les cas où les experts humains ne pouvaient fournir que des règles floues partielles ou incomplètes, lorsqu'un ensemble de problèmes ou de données d'entrée-sortie du système étaient disponibles, il a été jugé préférable d'extraire des relations floues ou des règles floues à partir des données système et de combiner les données, dans la mesure du possible, avec les connaissances et l'expérience humaine. La combinaison pourrait alors être utilisée pour construire un ensemble complet et pertinent de règles floues.

Une application très tôt de la logique floue(LF) dans le diagnostic de défauts a été menée par (Goode et Chow, 1995), dans laquelle un détecteur hybride neuro-flou a été utilisé pour détecter les défauts du moteur. Dans cette étude, un détecteur de défauts neuro-flou a été utilisé pour surveiller l'état de l'usure des roulements du moteur et la défaillance de l'isolation de l'enroulement du stator. Les règles floues IF-THEN pour la classification de l'usure des roulements par le détecteur ont été construites de manière heuristique pour trois plages d'appartenance: faible, moyenne et élevée. Le détecteur de défaut neuro-flou entraîné était capable de fournir des résultats de détection de défauts précis et pouvait également fournir le raisonnement heuristique derrière le processus de détection de défauts et les conditions réelles des défauts du moteur.

Une autre application très tôt de la logique floue (LF) dans le diagnostic de défauts a été proposée par (Goddu et al, 1998) dans laquelle une méthode basée sur la logique floue a été utilisée pour interpréter des signaux vibratoires d'un moteur électrique afin de diagnostiquer les défauts de roulement. Les données du spectre du signal de vibration ont été entrées dans le système de décision flou et un résultat de diagnostic de défauts valide a été obtenu. Il a été suggéré que l'incorporation de réseaux de neurones ou les algorithmes génétiques avec la logique floue afin d'améliorer les capacités du système de décision floue serait bénéfique.

(Vicente et al, 2001) ont présenté un travail sur un système de diagnostic automatique pour la détection et la classification des défauts de roulements à l'aide de la logique floue. Les signaux vibratoires mesurés ont été analysés en utilisant des techniques spectrales et statistiques. Les variables utilisées en tant qu'entrées pour le système flou comprenaient: la charge radiale, la vitesse de l'arbre, le kurtosis, le skewness et le RMS. Le système conçu a permis de classer trois types de défauts préétablis de roulements qui fonctionnaient sous plusieurs vitesses de rotation d'arbre et conditions de charge. Les résultats ont montré que le système conçu était capable de diagnostiquer 97% de la base de données de test pour faire la distinction entre les états normaux et les états avec défauts. Le système a atteint une précision de 95% dans la classification des défauts des cas ; corrosion normale, corrosion avec des creux et corrosion avec des rayures.

(Miguel et Blazquez, 2005) ont appliqué la logique floue dans une application de diagnostic basé sur un modèle pour un contrôleur de moteur à courant continu (DC motor). La logique floue a été utilisée pour traiter l'incertitude du modèle de système, le bruit et d'autres variables qui réduisent la fiabilité et la robustesse de la méthode de diagnostic des défauts. Le système de détection et d'isolation des défauts à base de la logique floue(LF) a été appliqué avec succès dans un laboratoire où l'incertitude causée par les perturbations et les erreurs de modélisation ont été réduites.

(Celik et Bayir, 2007) ont étudié l'application d'un système complémentaire à base de la logique floue pour le diagnostic de défauts d'un moteur à combustion interne. Les règles floues du système ont été construites en utilisant des connaissances théoriques, des connaissances expertes et les résultats expérimentaux. La précision du classifieur flou a été testée par des études expérimentales et qui ont été effectuées sous différentes conditions de défauts. En utilisant le système de diagnostic de défauts développé, dix défauts généraux qui ont été observés dans le moteur à combustion interne ont été diagnostiqués avec succès en temps réel.

(Wu et Hsu, 2009) ont étudié le développement d'un système d'identification des défauts d'engrenage en utilisant des signaux vibratoires avec l'inférence en logique floue et la transformée en ondelettes discrètes (discret wavelet transform(DWT)) pour un système expérimental d'engrenages. Le système proposé qui utilisait la combinaison de la technique de

transformé en ondelettes discrètes pour l'extraction des paramètres du signal et l'inférence en logique floue pour l'identification de défauts a été étudié L'inférence floue a été proposée pour développer les règles de diagnostic de la base de données dans le système d'identification des défauts. Les travaux expérimentaux ont été effectués pour évaluer l'effet du diagnostic de défauts dans un système à engrenages sous diverses conditions de fonctionnement. Les résultats expérimentaux ont montré que le système de diagnostic de défauts proposé était efficace dans la mesure où il a augmenté la précision de l'identification des défauts d'engrenage du système d'engrenages. Le but d'utilisation de la logique floue (LF) était de surmonter les difficultés dans le diagnostic des défauts des machines tournantes dans un environnement complexe et bruité et de réduire le besoin de connaissances d'un technicien expérimenté

(Saravanan et al, 2009) ont utilisé un classifieur flou obtenu à partir de l'information intuitive et des connaissances de domaine relatif aux caractéristiques de défauts pour un système de diagnostic de défauts d'engrenage conique. Un arbre de décision a été utilisé pour sélectionner les meilleurs paramètres statistiques qui pourraient distinguer entre les différents états de défauts à partir des signaux vibratoires de la boîte de vitesses. Il a été suivi par la génération d'un ensemble de règles à partir des paramètres extraits puis attribué à un classifieur flou pour le processus de classification des défauts. Les résultats du classifieur flou ont été jugés encourageants.

(Umbrajaakar et Krishnamoorthy, 2018) ont présenté une méthode pour la prédiction des défauts de désalignement dans les machines tournantes à partir de leurs signaux vibratoires. Cette méthode est basée sur une approche combinée de la transformée en ondelettes discrètes (DWT : Discrete Wavelet Transform) et la logique floue. La DWT aide à la décomposition à plusieurs niveaux du signal vibratoire pour comprendre où et à quel moment le défaut de désalignement se produit. Cela implique la sélection de l'ondelette mère. Le FIS (Fuzzy Inference System) est combiné au DWT pour prédire le degré de défaut de désalignement présent. L'inclusion du système FIS a amélioré la précision de la prévision du degré de désalignement, qui a déclaré une erreur inférieure à 1%.

2.4 Hybridation des Techniques d'intelligence artificielle (IA) dans le diagnostic des défauts

La combinaison de deux ou plusieurs techniques d'intelligence artificielle (IA) pour construire un modèle hybride est basé sur l'objectif d'intégrer simultanément les avantages de chaque approche individuelle et de surmonter leurs faiblesses ou leurs insuffisances. La combinaison la plus courante trouvée dans les systèmes d'application de diagnostic de défauts combine les techniques des réseaux de neurones artificielles(RNA) et la logique floue(LF). Il forme un système hybride qui utilise une combinaison de RNA et LF, ce qu'on appelle un système neuro-flou.

2.4.1 Système neuro-flou

Un système neuro-flou se réfère au système par lequel diverses techniques d'apprentissage sont utilisées à l'aide des réseaux de neurones qui sont intégrées avec un modèle de système d'inférence floue. Le système vise à utiliser les avantages des réseaux de neurones et de la logique floue afin de fournir une initialisation précise du réseau de neurone en termes du système de raisonnement flou (Zhao et al, 2009).

L'objectif principal de l'intégration des réseaux de neurones et des systèmes flous est de combiner les avantages des deux méthodes afin de permettre un apprentissage adaptatif dans les systèmes de diagnostic avec une représentation transparente de connaissances (Leonhardt et Ayoubi, 1997).

La mise en œuvre des avantages combinés de la logique floue et des réseaux de neurones offre au système un processus de diagnostic supérieur. En outre, L'avantage de la combinaison des caractéristiques fournit au détecteur de défaut une caractéristique d'adaptabilité qui donne une plus grande précision à la solution dans des différents états de fonctionnement (Altug et al, 1999).

L'intégration vise à utiliser la fonctionnalité de logique floue pour traiter des informations imprécises provenant de conditions imprécises, et à fournir les exigences de modélisation initiales pour les systèmes de réseaux neuronaux qui ont la capacité d'entraînement, et peut effectuer des calculs en parallèle et avoir une tolérance au bruit. L'intégration fournit un

système de diagnostic de défauts robuste qui combine des informations numériques (quantitatives) et symboliques (qualitatives) (Patton et al. 2000).

2.4.1.1 Application du système neuro-flou dans le diagnostic de défauts

(Garga et al, 2001) ont introduit une méthode de raisonnement hybride qui intégrait les données de machines dans un réseau de neurones à propagation avant (a feed-forward neural network). Le système a utilisé le processus d'apprentissage pour obtenir une représentation de la connaissance du domaine explicite pour le pronostic de défauts d'une boîte de vitesses. La méthode a combiné les réseaux de neurones et les systèmes d'inférence floue qui ont construit un prédicteur neuro-flou. En utilisant cette combinaison, la capacité du système d'inférence floue à utiliser des descriptions linguistiques est renforcée par les procédures d'apprentissage utilisés par les réseaux de neurones.

(Wang et al, 2004a) ont été utilisés un système neuro-flou pour évaluer l'état des engrenages à denture droite au cours du fonctionnement. La structure du système d'inférence floue (FIS) a été élaborée à l'aide des connaissances des experts, et une procédure d'apprentissage au réseau de neurones a été utilisé pour générer les fonctions d'appartenance floue associées. Les résultats du test ont démontré que le système neuro-flou proposé améliorerait significativement la précision du diagnostic en raison de son adaptabilité et de sa robustesse.

(Wang et al, 2004b) ont présenté Les résultats d'une comparaison entre les réseaux de neurones récurrents (recurrent neural network(RNN)) et les systèmes d'inférence neuro-floue qui ont été utilisés pour prédire les tendances de propagation de défauts. Il a été constaté qu'un système neuro-flou (NF) entraînerait correctement se comporte mieux que les RNN dans la précision des prévisions et l'efficacité du processus d'entraînement(apprentissage).

(Chinnam et Baruah, 2004) ont présenté une application d'un système neuro-flou pour estimer la durée de vie utile restante (the remaining useful life(RUL)) d'un outil de forage. Dans l'étude, l'existence d'une définition spécifique du modèle concernant la défaillance et les données sur la défaillance n'étaient pas disponibles pour le système étudié. Cependant, il y'a une nécessité à un système neuro-flou en parallèle avec les connaissances et l'expérience d'un expert du domaine pour construire un modèle estimateur. Le système estimateur a été entraîné

à l'aide d'informations provenant d'experts du domaine ayant une connaissance adéquate sur le processus. Le modèle estimateur a été construit à l'aide d'un système neuro-flou et il a donné de bons résultats lors de l'estimation de la vie utile restante (RUL) de l'outil de forage.

(**Satish et Sarma, 2005**) ont proposé une combinaison de réseaux de neurones et de logique floue pour former un réseau neuro-flou avec rétropropagation (fuzzy back-propagation network) afin d'identifier l'état actuel d'un roulement et d'estimer la durée de vie utile restante (RUL) d'un moteur électrique. Dans leur travail, les résultats du réseau neuro-flou avec rétropropagation ont été comparés à ceux d'un réseau de neurones générique. L'étude a montré que l'approche hybride était préférable pour évaluer l'état actuel du roulement et le temps disponible avant que le remplacement soit nécessaire.

(**Kothamasu et Huang, 2007**) ont été présentés Une approche de modélisation neuro-floue, dans laquelle le système est basé sur l'apprentissage adaptatif avec l'utilisation d'un modèle flou de type Mamdani pour le diagnostic et le pronostic d'un système. Le système a été conçu pour fonctionner comme un outil d'aide à la décision pour aider à la maintenance conditionnelle. Il a eu une caractéristique permettant une adaptation pour fournir une amélioration continue grâce à l'interaction avec les utilisateurs.

(**Castejon et al, 2010**) ont mis en œuvre une combinaison de systèmes LF et RNA pour former un système intégré neuro-flou visant à traiter des problèmes complexes de classification de défauts. Ce système a utilisé la relation entre un ensemble de formes et les types de défauts sans qu'il soit nécessaire de modéliser explicitement les processus internes. Le système combiné s'est bien comporté en ce qui concerne le traitement des informations incertaines, ce qui est très similaire au processus de raisonnement humain. Le RNA a été utilisé car il a des capacités d'apprentissage en temps réel, de calcul parallèle et d'auto-organisation. La logique floue a été choisie en fonction de sa capacité à traiter des informations imprécises ou inexacts et qui est utile au traitement de tâches de classification vagues ou imprécises.

2.4.2 Système d'inférence neuro-flou adaptatif (ANFIS)

Le système d'inférence neuro-flou adaptatif (ANFIS) est une variante des systèmes basés sur les réseaux de neurones qui est couramment utilisé dans les recherches de diagnostic des

défauts de roulements. Il a été introduit pour la première fois par **(Jang, 1993)**. C'est l'un des systèmes hybrides intelligents les plus utilisés et il offre les avantages de la logique floue et des réseaux de neurones. Il utilise un ensemble de données (entrée-sortie) donné pour construire un système d'inférence floue (FIS), dont les paramètres des fonctions d'appartenance sont ajustés en utilisant soit un algorithme de rétropropagation, soit en combinaison avec une méthode de type des moindres carrés. Ceci permet au système flou d'apprendre à partir des données utilisées et cela pour le but de modélisation.

La technique d'apprentissage neuro-adaptative incorporée dans l'ANFIS est basée sur une méthode par laquelle la procédure de modélisation floue est capable d'apprendre des informations sur l'ensemble de données. Il calcule ensuite les paramètres de la fonction d'appartenance qui modélise l'ensemble de données (entrée-sortie) donné.

En termes de logique floue, il fournit une fonctionnalité de mappage entre les entrées et les sorties. Le processus de mappage est réalisé en utilisant les fonctions d'appartenance (MF) de l'entrée et ses paramètres associés en relation avec les fonctions d'appartenance de la sortie. C'est un outil utilisé pour accomplir la création et l'ajustement des fonctions d'appartenance de la logique floue et en combinaison avec la fonction d'apprentissage des RNA.

ANFIS a formulé un modèle d'inférence floue de type Sugeno (Sugeno fuzzy inference model (SFIM)), qui a été à l'origine proposé par **(Takagi et Sugeno, 1985) et (Sugeno et Kang, 1988)**. Le SFIM est un cas particulier du système d'inférence floue de Mamdani dans lequel le conséquent de chaque règle est définie par un singleton flou (ou un conséquent pré-fuzzifié) **(Jang et al, 1997)**. Ce dernier est intégré dans le cadre d'un RNA multicouche dans lequel les poids synaptiques (la force de connexion entre deux neurones) ne sont pas utilisés. Au lieu de cela, il utilise des nœuds adaptatifs et non adaptatifs **(Reddy et Mohanta, 2007)**.

La fonctionnalité de l'ANFIS est équivalente à celle du SFIM. Comme SFIM, ANFIS utilise un ensemble de paires de données d'apprentissage entrée-sortie pour régler la fonction d'appartenance et d'autres paramètres associés en utilisant une méthode de rétro-propagation du gradient ou une méthode des moindres carrés. ANFIS représente une intégration d'un modèle flou de type Sugeno où l'optimisation du système d'inférence floue final est obtenue grâce à l'apprentissage avec les RNA.

ANFIS est utile dans les cas où il y a un besoin d'appliquer le système FIS à un système pour lequel une collection de jeux de données entrées-sorties est disponible et les données sont destinées à être utilisées pour la modélisation, le suivi de modèle ou des besoins similaires.

En pratique, il existe quelques cas de modélisation de diagnostic de défauts dans lesquelles la construction des fonctions d'appartenance à base de la logique floue peut être effectuée facilement seulement en discernant manuellement les données disponibles. Il y a aussi des cas où la détermination manuelle des fonctions d'appartenance à base de la logique floue n'est pas préférée en raison de la complexité des relations entrée-sortie de l'ensemble de données. Dans ces cas, la sélection des paramètres associés à une fonction d'appartenance donnée est principalement effectuée par des procédures d'essai et d'erreur. En utilisant ANFIS, une structure prédéterminée du modèle basée sur les caractéristiques des variables du système conçu n'est pas nécessaire.

La méthode préférée consiste à utiliser un processus de sélection des paramètres de manière dont on peut créer automatiquement les fonctions d'appartenance, en fonction des relations de données entrée-sortie. Cette méthode est également capable de s'adapter aux variations qui existent à l'intérieur des données. C'est là qu'une technique d'apprentissage neuro-floue telle que ANFIS est utile.

L'expertise d'un opérateur humain, ou d'experts, peut être utilisée sur le système modélisé pour générer les fonctions d'appartenance initiales et les règles pour le système d'inférence floue (FIS). L'ANFIS est ensuite utilisé pour affiner les règles IF-THEN et les fonctions d'appartenance floues initiales afin de mieux correspondre à la relation entre les caractéristiques de l'entrée et de la sortie du système ou des données (Lou et Loparo, 2004).

Cette caractéristique fournit la capacité d'utiliser des ensembles de données d'apprentissage (entrée-sortie) prédéterminés qui emploient ANFIS, dans la régulation des fonctions d'appartenance et d'autres paramètres associés au moyen de méthodes de rétro-propagation du gradient et de moindres carrés. Ces méthodologies fournissent une plus grande objectivité dans l'utilisation du modèle ANFIS puisqu'elles fournissent un moyen plus systématique qui dépend moins de la connaissance experte, ce qui est utile dans le diagnostic des défauts (Zhang et al, 2010).

2.4.2.1 Architecture d'un système ANFIS

Le système ANFIS est un réseau adaptatif proposé par Jang en 1993. Ce système peut être vu comme un réseau de neurones non bouclé pour lequel chaque couche est un composant d'un système flou. Le modèle ANFIS est le modèle le plus utilisé en pratique. L'architecture du système ANFIS affine les règles floues obtenues par des experts humains et réajuste le chevauchement entre les différents SEF (sous ensemble flou) pour décrire le comportement entrée-sortie d'un système complexe en utilisant une base de données pour l'apprentissage. Il s'agit d'une technique Neuro-Floue (NF) hybride qui apporte les capacités d'apprentissage des réseaux de neurones au système d'inférence flou de type Takagi-Sugeno. Le rôle de l'apprentissage est l'ajustement des paramètres de ce système d'inférence flou (partie prémisse et partie conclusion des règles). Le système hybride NF de type ANFIS est composé de cinq couches où les nœuds adaptatifs sont situés à la première et la quatrième couche. Afin de présenter l'architecture de base et le fonctionnement du système ANFIS qui sera utilisé dans notre travail, Le réseau adaptatif ANFIS est un réseau multicouche dont les connexions ne sont pas pondérées, ou ont toutes un poids de 1. Les nœuds sont de deux types différents selon leur fonctionnalité : les nœuds carrés (adaptatifs) contiennent des paramètres, et les nœuds circulaires (fixes) n'ont pas de paramètres. Toutefois, chaque nœud (carré ou circulaire) applique une fonction sur ses signaux d'entrées.

La Figure 2.1 présente le principe de raisonnement flou implémenté dans un système ANFIS, la Figure 2.2 illustre la structure des couches du système ANFIS. Les Figures 2.1 et 2.2 sont utilisées pour fournir une brève explication sur ANFIS. Pour simplifier, un exemple sur ANFIS comportant deux entrées x et y et une sortie f est utilisé dans le but d'expliquer sa structure. Il est supposé que la base de règles ne contient que deux règles if-then du modèle d'inférence floue de type Sugeno du premier ordre.

La Figure 2.1 illustre le principe de raisonnement flou avec deux entrées et une seule sortie. L'architecture ANFIS équivalente correspondante est illustrée à la figure 2.2. Les fonctions des nœuds et des couches sont expliquées ci-dessous:

Le concept de structure ANFIS peut être représenté à l'aide d'une base de règles simple, définie comme suit:

règle 1: si x est A_1 et y est B_1 Alors $f_1 = p_1x + q_1y + r_1$,

règle 2: si x est A_2 et y est B_2 Alors $f_2 = p_2x + q_2y + r_2$,

Où x et y sont les entrées, et A_1, A_2, B_1 et B_2 sont les ensembles flous qui représentent des valeurs linguistiques telles que petit, moyen, grand. Ces ensembles flous seraient déterminés au cours du processus d'apprentissage. $p_1, q_1, r_1, p_2, q_2, r_2$ Sont des paramètres de conception également déterminés au cours du processus d'apprentissage.

Couche 1:

Les nœuds de cette couche représentent des nœuds d'entrée et sont également appelés nœuds adaptatifs. Le niveau d'appartenance est généré par les nœuds de cette couche à l'aide de fonctions:

$$O_{1i} = u_{A_i}(x) \quad i = 1,2$$

$$O_{2i} = u_{B_i}(y) \quad i = 1,2$$

Où O_{1i} et O_{2i} sont les niveaux d'appartenance flous utilisés pour spécifier le degré d'appartenance des entrées nettes x et y en termes de valeurs linguistiques A_i , et B_i . A_i et B_i sont des ensembles flous construits à l'aide de leurs fonctions d'appartenance u_{A_i} et u_{B_i} . La forme sélectionnée des fonctions d'appartenance est normalement la forme de cloche (bell-shaped) avec un maximum égal à 1 et un minimum égal à 0, comme il est défini dans l'équation 2.1.

$$u_{A_i}(x) = \frac{1}{1 + \left| \frac{x - c_i}{a_i} \right|^{2b_i}} \quad (2.1)$$

Ou sous forme gaussienne avec la fonction :

$$u_{A_i}(x_1) = \exp \left[- \left(\frac{x_1 - c_i}{a_i} \right)^2 \right] \quad (2.2)$$

Dans le modèle de (Jang 93), les fonctions d'appartenances sont des gaussiennes,

Avec a_i , b_i et c_i paramètres de la fonction d'appartenance qui déterminent la forme de la fonction d'appartenance basée sur des valeurs linguistiques. Ces paramètres sont dans la partie prémisse des règles if – then et sont donc appelés paramètres de prémisse.

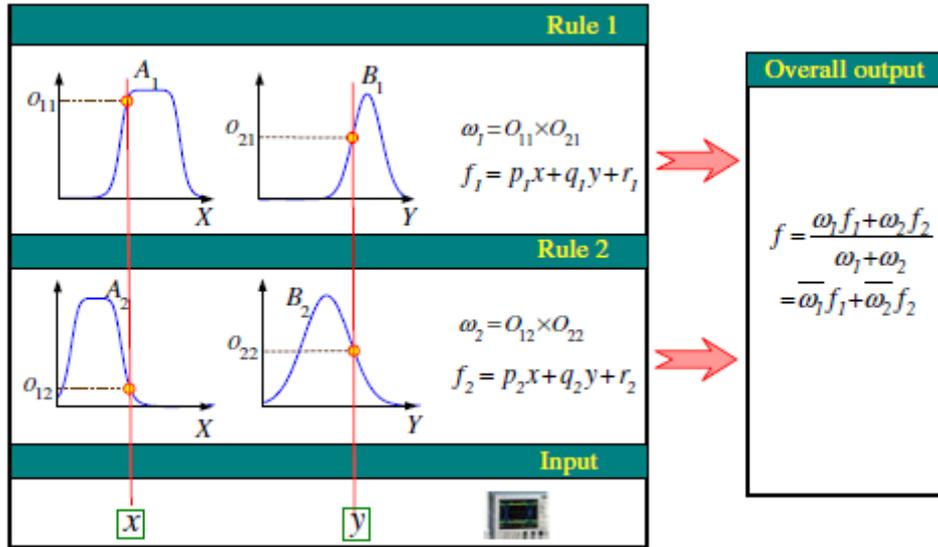


Figure 2.1 Principe de raisonnement flou (Modèle d'inférence floue de Sugeno) (Jang, 1993), (Zhang et al. 2010)

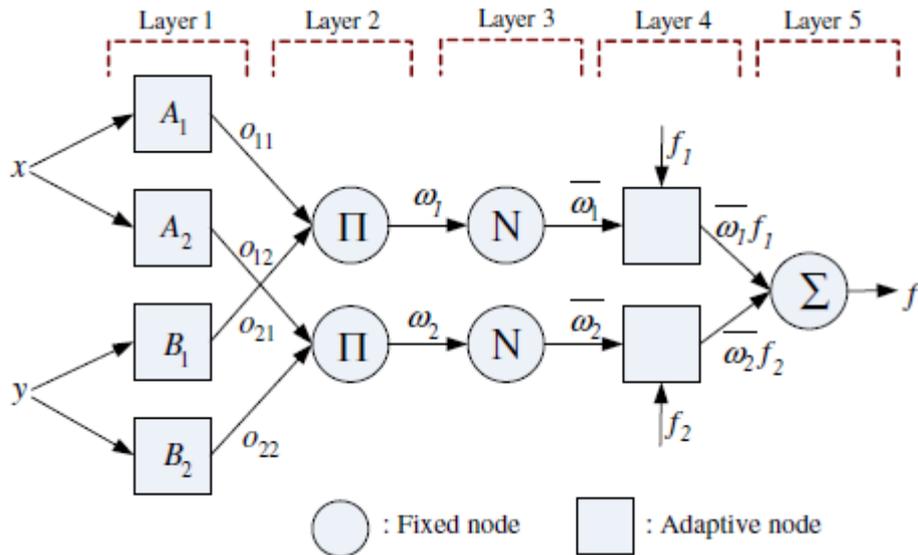


Figure 2.2 structure des couches ANFIS (Zhang et al. 2010)

Couche 2:

Chaque nœud de cette couche est un nœud fixe portant l'étiquette Π et qui multiplie les signaux entrants et génère le produit w_i . Par exemple,

$$w_i = u_{A_i}(x) \times u_{B_i}(y) \quad (2.3)$$

La sortie de chaque nœud de cette couche représente le poids d'une règle.

Couche 3:

Cette couche contient des nœuds fixes étiquetés N qui permettent de calculer le rapport entre le poids de la règle et la somme des poids de toutes les règles, également ce rapport \bar{w}_i est appelé le poids normalisé de la règle:

$$\bar{w}_i = \frac{w_i}{w_1 + w_2} \quad i = 1,2 \quad (2.4)$$

Couche 4:

Cette couche contient les nœuds adaptatifs. La sortie de chaque nœud est simplement le produit du poids normalisé avec un polynôme de premier ordre (pour un SFIM (sugeno fuzzy inference model) du premier ordre). Les sorties de la couche 4 sont définies par:

$$\bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \quad i = 1,2 \quad (2.5)$$

Où \bar{w}_i est la sortie de la couche 3 et p_i, q_i et r_i sont des paramètres linéaires du SFIM du premier ordre, qui sont appelés paramètres consécutifs.

Couche 5:

Il n'y a qu'un seul nœud dans cette couche. C'est un nœud fixe qui est étiqueté Σ . Comme son étiquette l'indique, il effectue la somme de tous les signaux entrants. Par conséquent, la sortie globale du modèle est définie par:

$$f = \text{sortie globale} = \sum \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}, \quad i = 1,2 \quad (2.6)$$

Les cinq couches expliquées ci-dessus fonctionnent comme un SFIM du premier ordre dans lequel il existe deux couches adaptatives, à savoir la première et la quatrième couche. La première couche contient les paramètres prémisses (a_i, b_i, c_i) et la quatrième couche contient les paramètres conséquents (p_i, q_i, r_i). Ces paramètres sont obtenus via un algorithme d'apprentissage hybride. L'algorithme utilisé est une combinaison de l'approche de descente de gradient et de l'estimation par les moindres carrés, L'algorithme comprend deux étapes afin d'améliorer l'efficacité de l'apprentissage et d'éliminer le piègeage possible dû aux minima locaux.

Dans la première étape, les paramètres prémisses sont supposés être fixes. Les paramètres conséquents optimaux sont ensuite obtenus en utilisant l'estimation des moindres carrés. Dans la deuxième étape, les paramètres conséquents sont supposés être fixes. Les paramètres prémisses sont ensuite mis à jour à l'aide de la méthode de descente de gradient par rétro-propagation, basé sur les valeurs d'erreur.

2.4.2.2 Avantages de l'ANFIS

L'expertise d'opérateurs humains ou d'experts sur le système modélisé ou les données peut être utilisé pour générer des fonctions d'appartenance et des règles initiales pour le système d'inférence floue. L'ANFIS est ensuite utilisé pour affiner les règles initiales floues if-then et la fonction d'appartenance pour mieux correspondre à la relation entre les caractéristiques d'entrée et de sortie d'un système complexe ou d'un ensemble de données (Lou et Loparo, 2004).

En utilisant des ensembles de données d'entraînement (entrée-sortie) prédéterminés, un système ANFIS peut être utilisé pour régler la fonction d'appartenance et d'autres paramètres associés en utilisant la méthode de descente de gradient du rétro-propagation et la méthode des moindres carrés. Ces méthodologies fournissent une meilleure objectivité dans l'utilisation du modèle ANFIS, car elles fournissent un moyen plus systématique et dépend moins de la connaissance d'un expert, ce qui donne plus d'objectivité aux problèmes (Zhang et al, 2010)

2.4.2.3 Application du système ANFIS dans le diagnostic de défauts

L'idée d'utiliser l'ANFIS dans le diagnostic de défauts est d'obtenir une méthode de diagnostic plus robuste pouvant intégrer plusieurs paramètres bruts générés à partir des données.

Il existe différents systèmes de diagnostic de défauts qui utilisent l'ANFIS pour automatiser l'identification de l'état final des défauts et plusieurs exemples sont présentés ci-dessous.

(Zhang et Morris, 1996) ont présenté un modèle récurrent neuro-fuzzy pour la prévision à long terme, très similaire à l'ANFIS, dans lequel les fonctions étaient des modèles autorégressifs linéaires localement.

(Zhang et Morris, 1996) ont présenté un modèle neuro-flou récurrent pour la prévision à long terme, très similaire à ANFIS, dans lequel les fonctions des sorties étaient des modèles autorégressifs localement linéaire. Les modèles flous ont été implémentés par les réseaux de neurones flous qui combinent la capacité du raisonnement flou dans le traitement des informations incertaines et la capacité des réseaux de neurones à apprendre à partir des exemples. Les ensembles flous ont été utilisés pour représenter les anomalies du processus et qui ont ensuite été utilisés dans une approche floue pour représenter les anomalies du processus rendant le système de diagnostic plus résistant au bruit de mesures.

(Altmann et Mathew, 2001) ont présenté une méthode permettant d'améliorer la détection des informations de diagnostic provenant de défauts des roulements à basse vitesse. Cette méthode est basée sur l'application d'un système d'inférence neuro-flou adaptatif utilisé pour sélectionner automatiquement les paquets d'ondelettes appropriés correspondant aux caractéristiques de défaut.

Un système neuro-flou a été utilisé par **(Wang et al, 2004a)** pour évaluer plusieurs défauts d'un train d'engrenages droits lors de fonctionnement. Le système neuro-flou a été utilisé pour identifier les types de défauts, tels que ; engrenage fissuré avec 20% et 40% de la largeur de la racine de dent, engrenage écaillé avec 20% et 50% de la surface de dent. Dans ce travail, la structure du système FIS a été élaborée à l'aide de connaissance experte, tandis que la procédure d'apprentissage par les réseaux de neurones a été utilisé pour générer les fonctions d'appartenance floues associées. À travers des tests expérimentaux, il a été constaté que le classificateur neuro-fuzzy développé produit des résultats prometteurs en raison de ses capacités d'adaptation et de sa robustesse.

(Zhang et al, 2010) ont étudié l'application de l'ANFIS et de l'entropie multi-échelles (multi-scale entropy(MSE)) aux fins d'extraction de paramètres et de reconnaissance de défauts dans le diagnostic des défauts dans les roulements d'un moteur électrique. Plusieurs échelles de MSE ont été calculées à partir des signaux de vibratoires. Il y avait cinq paramètres statistiques comme la valeur maximale, la valeur minimale, la valeur moyenne arithmétique, la valeur moyenne géométrique et l'écart type qui ont été calculés à partir des MSE et utilisés en tant que paramètres. Les paramètres ont été ensuite présentés à l'ANFIS pour la classification des défauts. Les résultats expérimentaux ont indiqué que le classifieur ANFIS pourrait obtenir des résultats de diagnostic plus élevés. Ainsi, l'approche proposée a eu la possibilité pour le diagnostic d'un défaut naissant dans les roulements.

(Chen et al, 2016) ont proposé une méthode pour le diagnostic des défauts dans les engrenages planétaires basée sur l'entropie floue de la décomposition moyenne locale (local mean Decomposition : LMD) et le système d'inférence neuro-flou adaptatif (ANFIS). Le signal de vibration d'origine est décomposé en six composantes PF (PF: product function) et un résidu à l'aide de la décomposition moyenne locale (LMD). Étant donné que les composants PF décomposés contiennent les informations principales sur les caractéristiques des défauts, l'entropie floue est utilisée pour refléter la complexité et l'irrégularité de chaque composant PF. Les entropies floues de chaque composant PF sont définies en tant qu'entrées du modèle ANFIS, les paramètres du modèle ANFIS sont ajustés de manière adaptative en fonction des données d'apprentissage. Enfin, les règles d'inférence floues sont déterminées et le modèle optimal ANFIS est obtenu. Les données de test sont utilisées pour tester le modèle ANFIS formé. Le taux global de reconnaissance des défauts atteint 88,8% et le taux de reconnaissance des défauts pour les engrenages présentant une usure atteint 96%. Par conséquent, la méthode proposée est efficace pour diagnostiquer les défauts des engrenages planétaires.

2.5 Conclusion

Dans ce chapitre, une revue des techniques d'intelligence artificielle (Artificial Intelligence (AI)) largement utilisée dans le diagnostic de défauts a été présentée. Les trois techniques d'AI les plus utilisées sont ; les réseaux de neurones artificiels (RNA), la logique floue(LF) et le neuro-flou(ANFIS).

Les types des RNA et une liste d'applications de cette dernière en diagnostic de défauts dans les machines tournantes ont été passés en revue en présentant plusieurs applications de différents RNA en fonction des groupes de composants de machines. Un contexte et une introduction au concept de logique floue ont été présentés. Les avantages et les inconvénients des techniques de logique floue dans les applications de diagnostic de défauts ont également été détaillés.

La dernière section du chapitre présentait la motivation et les besoins pour la mise en œuvre d'une technique d'intelligence artificielle (IA) hybride ou combinée pour une meilleure performance dans le diagnostic de défauts. La technique neuro-floue, qui est l'une des techniques d'intelligence artificielle (IA) les plus utilisées, a été discutée. Le système susmentionné est un système hybride conçu pour intégrer les avantages des RNA et de la LF à des fins du diagnostic des défauts.

CHAPITRE 3

DEMARCHE EXPERIMENTALE

Chapitre 3

Démarche expérimentale

3.1 Introduction

Par la recherche bibliographique présentée au chapitre 2, on s'est fait une idée générale sur les applications de la logique floue, les réseaux de neurones, les réseaux neuro-flous adaptatifs pour le diagnostic des défauts mécaniques dans les machines tournantes. A partir du milieu des années 90, les applications de la logique floue en maintenance des machines tournantes a commencé à apparaître progressivement avec l'utilisation des différentes méthodes de classification floues (Mechefske, 1992a), (Mechefske, 1992b), (Mechefske, 1998). Dans ces dernières années nous avons remarqué que de nombreux travaux utilisent des systèmes d'inférence floue (FIS: Fuzzy Inference Systems) en vue de diagnostic, Ces systèmes souffrent de plusieurs problèmes soit dans leur mise en œuvre ou dans la faiblesse de leurs résultats (voir section 2.3.1 du chapitre 2) et c'est la raison pour laquelle nous devons optimiser ces systèmes, qui d'ailleurs est sur quoi notre travail se focalise.

Dans ce chapitre, nous présenterons l'essentiel de la démarche expérimentale que nous suivrons dans ce travail, en commençant par les différentes étapes de la mise en œuvre du système d'inférence floue(FIS) ainsi que le système neuro-flou adaptatif (ANFIS : Adaptive Neuro-Fuzzy Inference System) que nous utiliserons pour l'optimisation paramétrique du système FIS. Ensuite, nous présentons le banc d'essais de roulements et les signaux vibratoires prélevés sur ce dernier. Ces signaux sont utilisés ensuite pour vérifier l'efficacité et la validité des modèles conçus pour diagnostiquer les défauts de roulements.

3.2 Mise en œuvre d'un système à inférence floue (FIS)

Il existe 3 étapes fonctionnelles dans la mise en œuvre d'un système d'inférence floue (FIS) pour le diagnostic des défauts mécaniques des machines tournantes, comme le montre le schéma (Bandemer, 1995): (Voir Figure 3.1)

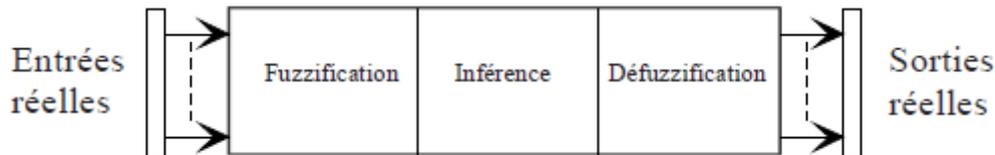


Figure 3.1 Étapes de la mise en œuvre d'un système à inférence floue (FIS)

3.2.1 Étape de fuzzification

L'étape de fuzzification consiste à caractériser les variables linguistiques utilisées dans le système. Il s'agit donc d'une transformation des entrées réelles en une partie floue définie sur un espace de représentation lié à l'entrée. Cet espace de représentation est décrite normalement par un sous-ensemble flou. Durant l'étape de la fuzzification, chaque variable d'entrée et de sortie est associée à des sous-ensembles flous. Pour chacune de ces variables, on doit connaître à priori son intervalle de définition, le nombre d'ensembles flous et les formes des fonctions d'appartenance associés à ces dernières.

3.2.2 Étape d'inférence

L'étape d'inférence consiste à utiliser le moteur d'inférence, qui est un mécanisme permettant de condenser l'information d'un système à travers un ensemble de règles définies pour la représentation d'un problème quelconque. Chaque règle délivre une conclusion partielle qui est ensuite agrégée aux autres règles pour fournir une conclusion (agrégation).

Après l'édition des règles, il reste à calculer les degrés d'appartenance des variables d'entrée et de sortie aux ensembles flous associés.

Chaque règle est composée de prémisses liées par les opérateurs ET, OU et donne lieu à une implication par l'opérateur ALORS.

L'agrégation de ces règles est une opération qui doit aboutir à une seule valeur de la variable de sortie après défuzzification.

3.2.3 Étape de défuzzification

Cette étape consiste à transformer la variable linguistique issue du système d'inférence flou en valeur numérique. Pour cela, il existe trois grandes méthodes (Meunie et Comarsal, 2003), (Zaddeh, 1976):

- La méthode du maximum qui correspond à l'abscisse minimale de l'ordonnée maximale de la surface. Elle est peu utilisée.

- La méthode de la moyenne pondérée.
- La méthode des centroides qui est la plus utilisée. Elle consiste tout simplement à calculer les centres de gravité des surfaces.

3.3 Déroulement du diagnostic des défauts par un système d'inférence floue (FIS)

La méthodologie choisie pour le diagnostic de défauts mécaniques dans les machines tournantes par un système à inférence floue (FIS) est représentée par l'organigramme de la Figure 3. 2.

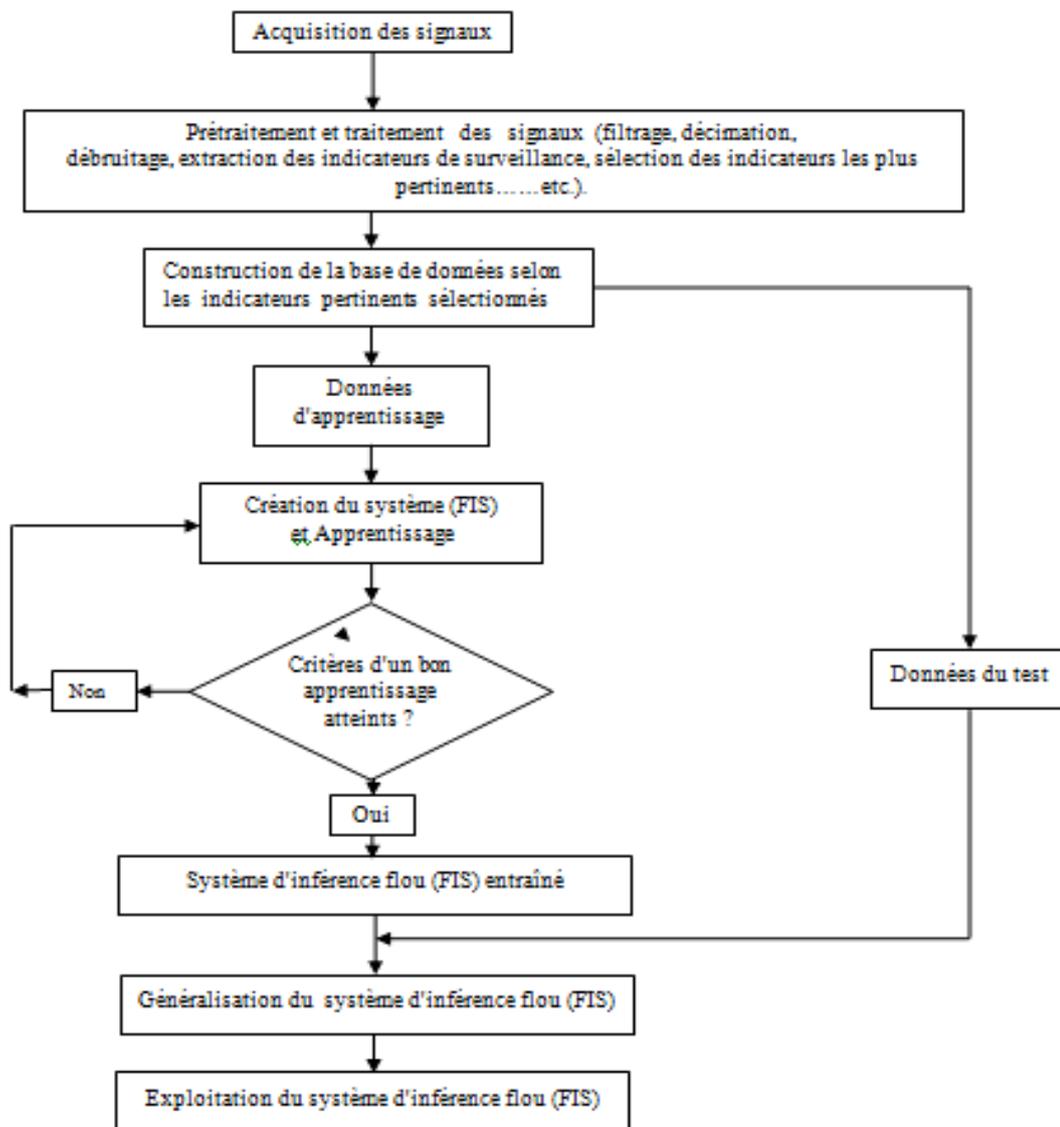


Figure 3. 2 Organigramme de mise en place du système d'inférence floue (FIS) pour la classification des défauts

3.3.1 Acquisition des signaux

Dans cette étape, on procède à une collecte (enregistrement) des signaux vibratoires correspondant aux différents états de fonctionnement du dispositif étudié. Cette opération se fait à l'aide d'une chaîne de mesure (capteurs, multiplexeur, amplificateur, convertisseur, carte d'acquisition, ...etc.) placée entre le dispositif en cours d'étude et l'unité de calcul (enregistrement et traitement).

3.3.2 Prétraitement des données

Cette étape est nécessaire pour éliminer toute sorte d'informations inutiles existantes dans la base de données (signaux). Ainsi, cette étape consiste à faire une préparation pour l'étape suivante (étape de traitement), c'est-à-dire passer les signaux vibratoires par un prétraitement qui contient les opérations de type (filtrage, décimation, débruitage, ...etc.).

3.3.3 Traitement des données

Cette étape consiste à faire des analyses, fréquentielle et temporelle du signal, c'est-à-dire l'extraction des indicateurs statistiques et fréquentiels. Aussi, on procède à une sélection d'indicateurs les plus pertinents parmi l'ensemble d'indicateurs disponible (dans notre cas, c'est l'ensemble des indicateurs de roulements).

3.3.4 Construction du vecteur forme

La sélection des indicateurs les plus pertinents dans l'étape précédente permet de former le vecteur de forme, ce dernier permet de représenter la base de données sous forme d'une matrice de dimension $(N \times M)$ où N est le nombre des observations de la base de données d'apprentissage où de test et M est le nombre des indicateurs sélectionnés.

3.3.5 Apprentissage et test du système d'inférence floue (FIS)

L'opération d'apprentissage consiste à chercher une configuration optimale des paramètres du système correspondant à un taux de succès élevé. Une fois le (FIS) entraîné. On procède à un test avec la base de données test pour vérifier que notre système réagit correctement, on pourra ainsi déterminer le taux de succès ou de bonne classification. Après ces deux opérations, le système (FIS) est prêt à l'exploitation.

3.3.6 Exploitation du système d'inférence floue (FIS)

Une fois que la configuration optimale du (FIS) est déterminée, le système devient capable de classer de nouveaux individus non appris.

3.4 Implémentation d'un système d'inférence flou (FIS) sous Matlab

Les différentes étapes d'implémentation d'un système à inférence floue (FIS) pour le diagnostic sous environnement Matlab sont schématisées dans la Figure 3.3 (Mokhtari, 1998).

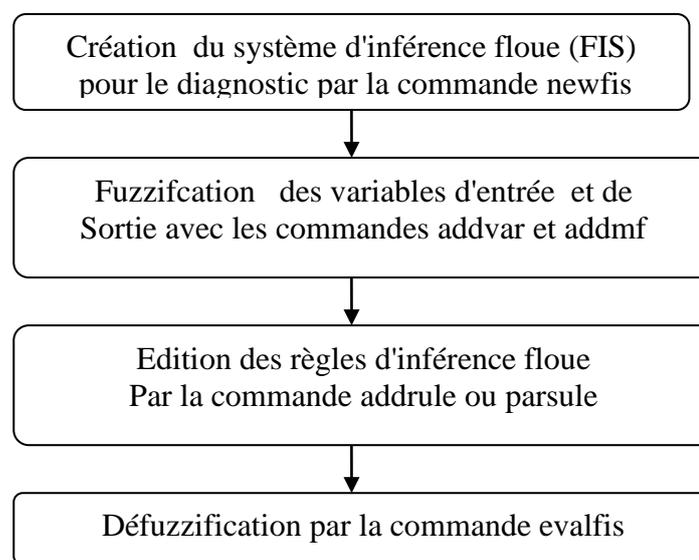


Figure 3.3 Étapes d'implémentation d'un système d'inférence floue (FIS) sous Matlab

3.4.1 Création du système d'inférence flou (FIS)

On commence dans cette étape par le choix du type de système qu'on veut implémenter, Soit le type Mamdani ou Sugeno.

La méthode de Mamdani est la première à avoir été proposée historiquement, elle date de 1975. Elle est beaucoup plus répandue dans la théorie du contrôle de procédés. La méthode de Sugeno (1985) en diffère au niveau de la définition de la variable de sortie et, par conséquent, des méthodes de Défuzzification.

Rien ne change au niveau de la fuzzification des variables d'entrée. La variable de sortie prend soit une valeur constante (singleton) indépendant des valeurs d'entrées, soit une combinaison

linéaire de celles-ci. De plus, chacune des deux méthodes, présente des caractéristiques et des avantages ([Bezd et Sank, 1992](#)); pour la méthode de Mamdani :

- Elle est intuitive ;
- Elle est largement utilisée ;
- Elle simule le raisonnement humain ;

Par ailleurs, la méthode dite Sugeno a les avantages suivants :

- Très efficace du point de vue calcul ;
- Elle fonctionne bien avec les techniques linéaires, l'optimisation et les techniques dites adaptatives.
- Enfin, elle privilégie l'analyse mathématique ;

➤ **Remarque:** Dans notre étude, nous avons choisi d'utiliser un système (FIS) de type Sugeno, par ce que ce type fonctionne bien avec les techniques linéaires et les techniques d'optimisation.

La création d'un système d'inférence flou (FIS) de diagnostic à l'aide des commandes Matlab se fait par la commande **newfis** qui accepte jusqu'à 7 arguments.

La syntaxe générale avec les 7 arguments est ([Mathworks, 2000](#)) :

**Sys_flou= newfis ('nom_syst','type','ET_method','OU_method','imp_method',
'agg_method', 'deffuz_method');**

nom_syst : nom du système flou,
 type : type Mamdani ou Sugeno,
 ET_methode : méthode utilisée pour l'opérateur ET (min, prod, custom) pour Mamdani et Sugeno,
 OU_method : méthode utilisée pour l'opérateur OU (max, probor, custom) pour Mamdani et Sugeno,
 imp_method : méthode d'implication pour Mamdani (min, prod où custom), pour Sugeno on n'a pas d'implication.
 agg_method : méthode d'agrégation des règles pour Mamdani (max, sum, probor, custom), pour Sugeno on n'a pas d'agrégation.
 deffuz_method : méthode de Défuzzification pour Mamdani (centroid, bisector, mom, lom, som, Custom), pour Sugeno on a les méthodes (wtaver, wtsum).

3.4.2 Fuzzification des variables d'entrées et de sorties

La définition des variables d'entrée et de sortie se fait par la commande **Addvar**:

Sys_flou = addvar ('nom_syst','type','nom','intervalle');

nom_syst : nom du système flou,
 type : variable d'entrée 'input' ou de sortie 'output',
 nom : nom de la variable auquel feront référence les règles flous,
 intervalle : intervalle de définition que prend la variable.

Après la définition des différentes variables d'entrée et de sortie par la commande **addvar**, les différentes fonctions d'appartenance associées aux différentes variables sont spécifiées par la commande **addmf**.

Sys_flou = addmf ('nom_syst','type','num','nom','interv', params);

nom_syst : nom du système flou,
 type : variable d'entrée 'input' ou de sortie 'output',
 nom : nom de la fonction d'appartenance (triangulaire, trapézoïdale, gaussienne,...).
 num : numéro de la variable (la variable n°1 est la 1^{ère} créée),
 interv : intervalle de valeurs que prend la variable,
 params : paramètres de la fonction d'appartenance (ex: moyenne et écart_type pour une fonction d'appartenance de type Gaussienne, etc.).

3.4.3 Édition de règles floues

Pour un système d'inférence floue possédant m variables d'entrée et n variables de sortie, l'ensemble des règles floues est défini par une matrice de règles possédant autant de lignes que d'ensembles flous de chacune des entrées et (m+n+2) colonnes.

La première règle floue constitue la première ligne de la matrice, la forme générale d'une règle avec m entrées et une seule sortie (conclusion) est la suivante:

SI [V_{e1} est A₁ ET V_{e2} est A₂ ET.....ET V_{em} est A_m] ALORS [V_{s1} est B₁]

Avec: V_{e1}, V_{e2},....., V_{em} : les m variables d'entrée du système flou (SIF)

V_{s1} : variable de sortie du système (une seule variable).

A₁: ensemble flou parmi les ensembles flous associés à V_{e1}.

.....

A_m : ensemble flou parmi les ensembles flous associés à V_{em}.

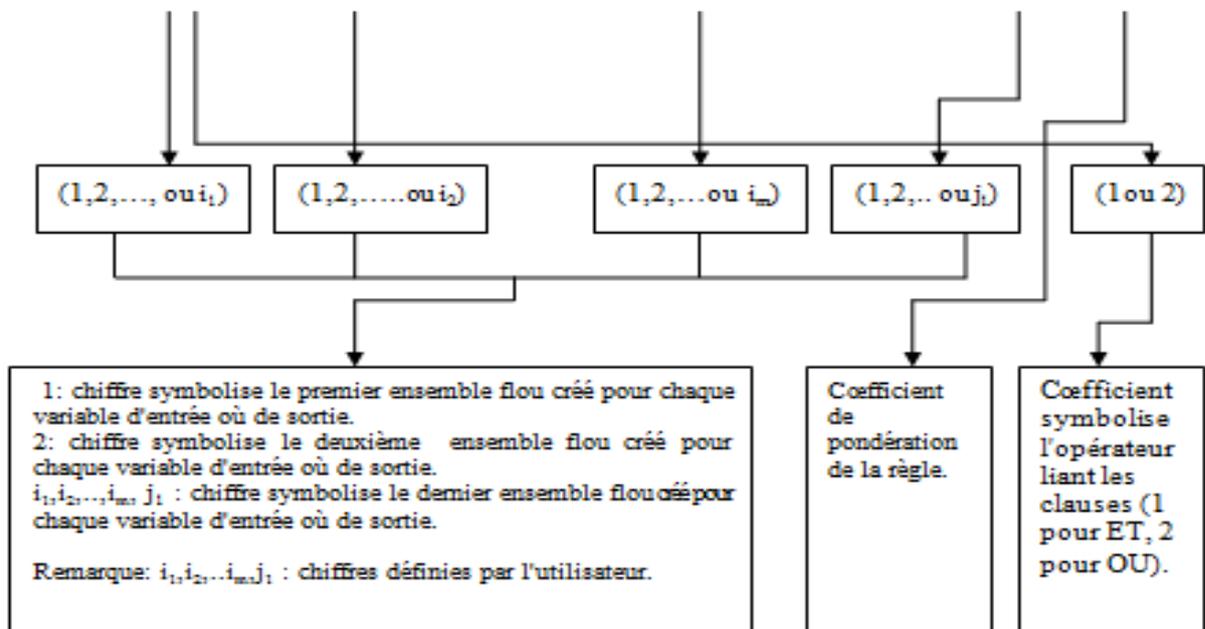
B₁ : ensemble flou parmi les ensembles flous associés à V_{s1}.

- le nombre des ensembles flous associés à chaque variable d'entrée ou de sortie est défini par l'utilisateur.

➤ **Remarques :**

- La matrice des règles, comme nous l'avons cité, possède autant de lignes que d'ensembles flous de chacune des variables d'entrées, et m+n+2 colonnes (pour notre cas, n=1(une seule sortie)), m : nombre de variables d'entrée.
- Pour l'édition des règles sous Matlab, On a deux possibilités, édition sous la forme symbolique (symbolic) et l'autre sous la forme textuelle (verbeuse).
- Dans la forme symbolique les éléments de la matrice sont des chiffres, et chaque chiffre symbolise un des éléments constituant les règles (ensembles flous associés aux différentes variables d'entrée et de sortie, les opérateurs liant les différentes clauses, un coefficient de pondération). Toujours dans cette forme, la matrice des chiffres est déduite à partir de la base de règles, la manière de déduction des lignes de la matrice à partir de la base des règles est la même pour l'ensemble des règles. Pour illustrer cette manière, on fait une déduction à partir de la forme générale des règles déjà donnée (voir ci-dessous).

$$\text{SI } [V_{e1} \text{ est } A_1 \text{ ET } V_{e2} \text{ est } A_2 \text{ ET } \dots \text{ ET } V_{em} \text{ est } A_m] \text{ ALORS } [V_{s1} \text{ est } B_1] \quad (1)$$



Donc, La ligne de la matrice déduite à partir de la règle générale est de la forme suivante:
 $[(1, 2, \dots, \text{ou } i_1) (1, 2, \dots, \text{ou } i_2) \dots (1, 2, \dots, \text{ou } i_m) (1, 2, \dots, \text{ou } j_1) (1 \text{ coefficient}) (1 \text{ ou } 2)]$

De la même manière, on obtient le reste des lignes de la matrice à partir de la base des règles.

- Une fois la matrice des chiffres définie, on procède à l'édition des règles, l'édition se fait par la commande **addrule**. La commande utilisée pour l'édition des règles pour la forme symbolique est :

Sys_flou = **addrule** ('nom_syst', regles) ;

Avec :

'nom_syst' : Nom du système d'inférence floue.

Regles : ensemble des règles.

- pour la forme textuelle (linguistique), l'édition des règles se fait sous forme d'une chaîne de caractères, la commande **parsrule** permet ensuite de l'ajouter sous le format adéquat à la matrice des règles. Par le choix de la langue, le français dans notre cas, la commande **parsrule** reconnaît les mots clés: SI, ALORS, Est, Et, Ou et NON. Il suffit de rentrer les différentes règles sous forme d'un vecteur de chaînes de caractères en faisant attention à rajouter des blancs à certaines d'entre elles pour qu'elles aient la même taille.
- La commande utilisée pour l'édition des règles pour cette forme est:

Sys_flou=parsrule ('nom_syst', regles,'verbos','français');

Avec :

Regles : ensemble des règles.

Verbos : forme d'édition des règles (forme textuelle).

Francais : la langue d'édition des règles (le français).

- La commande **showrule** affiche les règles de façon normalisée :

Showrule (sys_flou)

- On sauvegarde sous le même nom que celui du système flou préalablement défini à l'aide de la commande **writefis** :

Writefis (sys_flou,'nom_syst')

- Après que le système est complètement défini. La commande **getfis**, possédant plusieurs syntaxes, permet d'avoir des informations sur le système : nombre d'entrées, d'ensembles flous de chacune d'elles, etc.

Getfis (sys_flou,'name')

getfis (sys_flou,'input',1)

getfis (sys_flou,'output',1)

getfis (sys_flou)

`getfis (sys_flou,'input',1,'mf',1)`

3.4.4 Défuzzification

Lors de la fuzzification, pour chaque variable d'entrée réelle, on calcule ces degrés d'appartenance aux ensembles flous qui lui sont associés.

Dans l'étape de Défuzzification, on réalise l'opération inverse, à savoir, obtenir une valeur réelle de la sortie à partir des surfaces obtenues dans l'étape d'inférence.

Il existe plusieurs méthodes de Défuzzification, dont la plus utilisée est celle du centre de gravité.

La boîte à outil «fuzzy logic toolbox» dispose de plusieurs types de défuzzification (voir les arguments de la commande `newfis` section 3.4.1):

'centroid' : centre de gravité de la surface,

'bisector' : bissecteur de la surface,

'mom' : moyenne des maximas,

'som' : plus petit des maximas en valeur absolue,

'Lom' : plus grand des maximas en valeur absolue.

La commande `ruleview` affiche la fenêtre Rule viewer dans la quelle on peut Observer la défuzzification par la méthode choisie :

`ruleview(sys_flou)`

À l'aide de la souris on peut choisir des valeurs quelconques pour chacune des entrées et observer la fonction d'appartenance de la variable de sortie obtenue par la méthode choisie, par défaut la défuzzification est réaliser par la méthode du centre de gravité.

La commande `gensurf` (sys_flou) trace la surface de la variable de sortie en fonction des entrées dans une fenêtre graphique.

L'angle de vue de la surface obtenue par `gensurf` peut-être modifié par la commande `view` en spécifiant l'azimut(AZ) et l'altitude(EL) adéquats.

`AZ=45 ; EL=30 ; gensurf (sys_flou), view (AZ,EL),`

3.4.5 Test du système d'inférence floue (FIS)

Le test du système d'inférence floue (FIS) peut être obtenu par la commande `evalfis`. Si la matrice représentant le système d'inférence flou (FIS) n'est pas présente dans l'espace de travail, sa lecture se fait par la commande `readfis`.

`Sys_flou= readfis ('nom_syst') ;`

$X = [x_1, x_2, \dots, x_m]$; avec x_1, x_2, \dots, x_m : valeurs numériques des variables d'entrée

$Y = \text{evalfis}(X, \text{sys_flou})$

`ruleview` (`sys_flou`) , cette commande affiche la fenêtre **Rule viewer** dans laquelle on peut observer la défuzzification.

3.5 Implémentation d'un système ANFIS (Adaptive Neuro-Fuzzy Inference Systems) sous Matlab

3.5.1 Implémentation d'un système ANFIS par l'interface graphique (ANFIS editor GUI)

Les étapes d'implémentation d'un système (ANFIS) pour le diagnostic sous Matlab sont schématisées dans la figure (3.4).

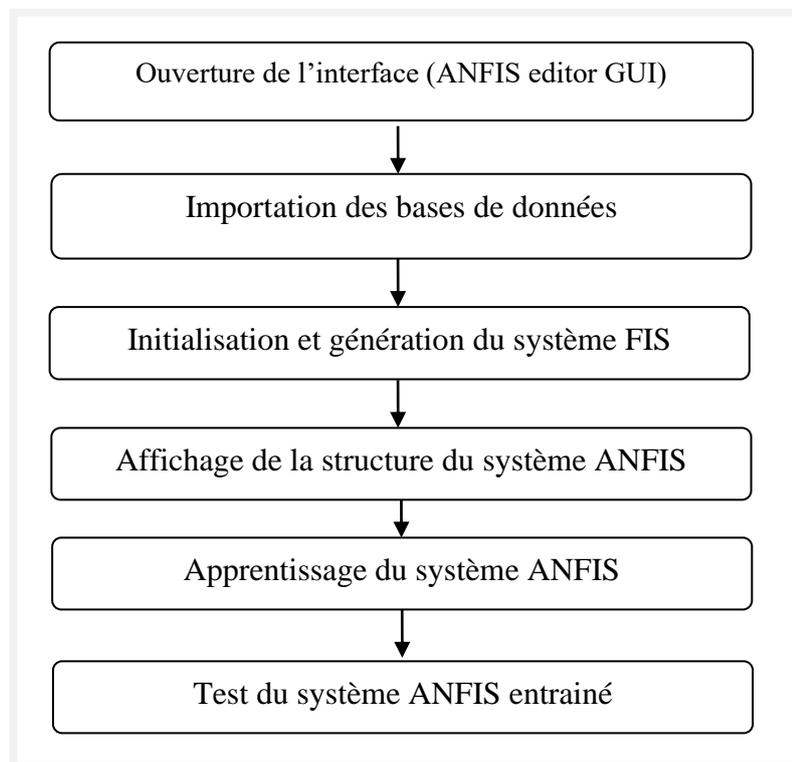


Figure 3. 4 Étapes d'implémentation d'un système (ANFIS) sous Matlab

3.5.1.1 Ouverture de l'interface (ANFIS editor GUI)

L'ouverture de l'interface graphique de l'éditeur ANFIS se fait par la commande `anfisedit` dans la ligne de commande MATLAB.

Une interface graphique (ANFIS editor GUI) s'affichera comme le montre la Figure 3.5. L'interface inclut quatre secteurs distincts pour soutenir un déroulement des opérations typiques. Le GUI vous laisse accomplir les tâches suivantes:

1. Chargez, tracez, et effacez les données
2. générez ou chargez la structure initiale du système FIS
3. apprentissage du système ANFIS
4. Valider le système ANFIS qualifié

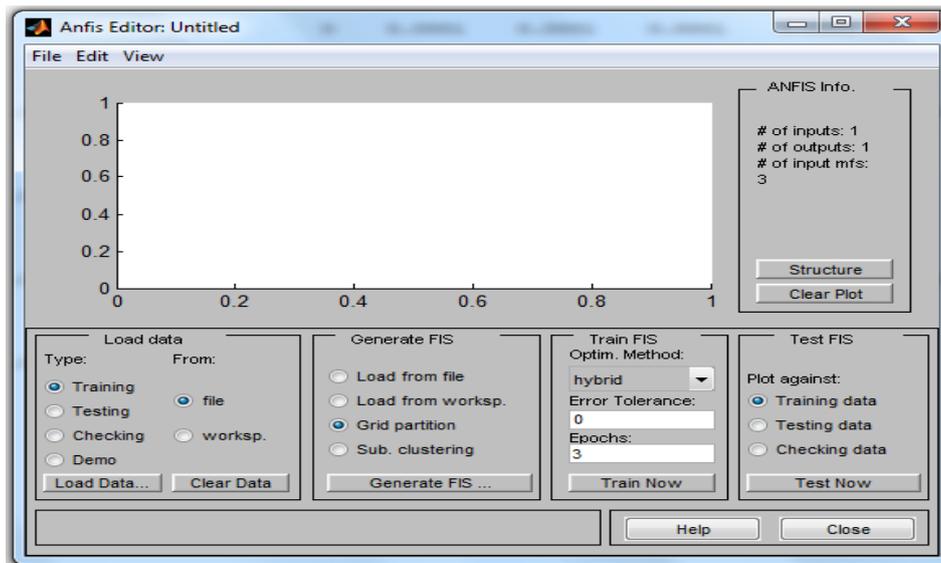


Figure 3.5 interface graphique pour l'implémentation d'un système ANFIS

3.5.1.2 Importation des bases de données

Dans la rubrique «load data» de l'interface graphique ANFIS, on sélectionne le type de la base de donnée qu'on veut importer (apprentissage(training) ; test(testing); vérification(checking) ;demo), ainsi que la source d'importation (espace de travail(worksp) ou dossier(file)), ensuite on clique sur «load data», une boîte de dialogue s'affiche (voir Figure 3.6), sur cette boîte on tape le nom de la base de donnée et on clique sur ok, la base de données apparaitre sur la partie centre(plot) de l'interface graphique , les observations sont représentées par des cercles O pour la base de données apprentissage et des plus(+) pour la base de données de vérification (voir Figure 3.7).

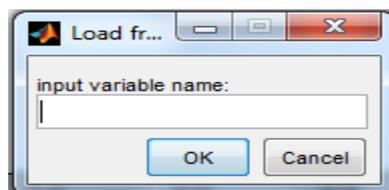


Figure 3.6 boîte de dialogue pour entrer le nom de la base de données

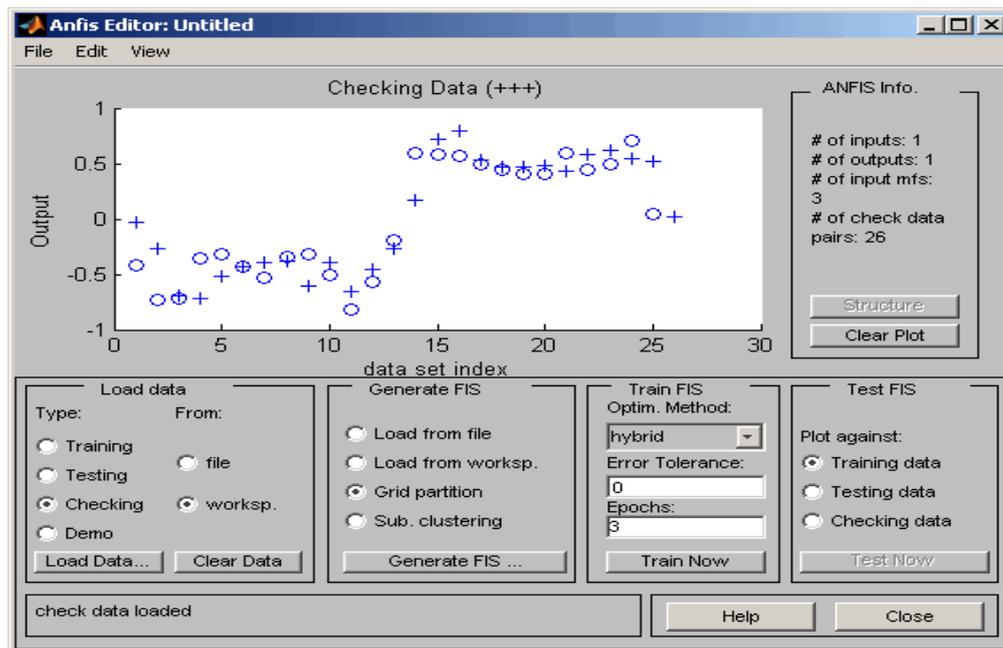


Figure 3.7 Affichage des bases de données

3.5.1.3 Initialisation et génération du système FIS

Il existe deux méthodes pour l'initialisation du système FIS :

- Génération automatique de la structure du système FIS
- Spécification de la structure FIS
- Pour la première méthode, la génération automatique de la structure du système FIS se fait comme suit :

Dans la rubrique «Generate FIS» de l'interface graphique ANFIS, on sélectionne une des deux méthodes ; «grid partition» qui est la méthode sélectionnée par défaut ou la méthode «sub clustering». Si on choisit par exemple la méthode «grid partition» et on clique sur «generate FIS», une fenêtre s'affiche (Figure 3.8) sur laquelle on peut choisir le nombre et le type des fonctions d'appartenance associées aux différentes variables d'entrée. Pour la fonction d'appartenance de la sortie on a deux choix, soit une fonction linéaire ou un singleton (fonction constante), et ce choix est dû au fait que le système ANFIS ne fonctionne qu'avec des systèmes FIS de type sugeno (Jang, 1993).

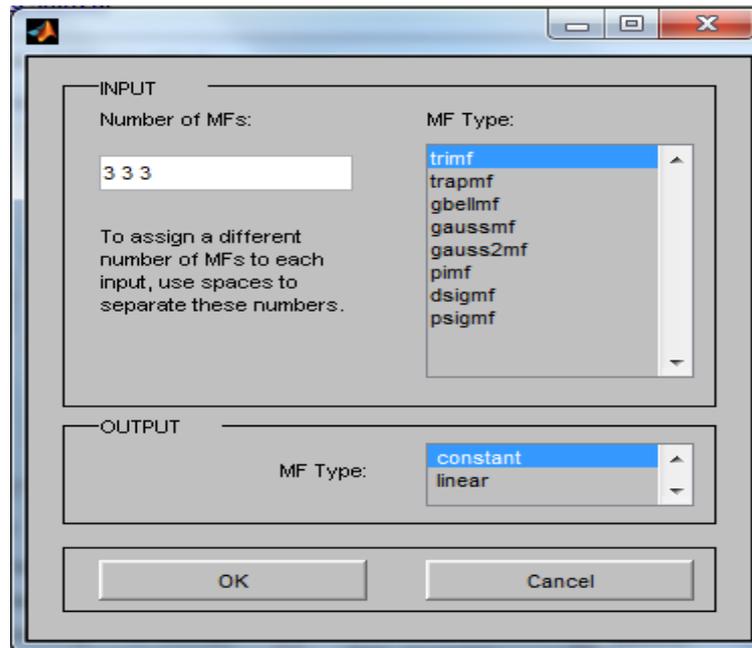


Figure 3.8 Initialisation et génération du système FIS

Après le choix des différentes paramètres on clique sur ok, et avec cela le système FIS initial sera généré.

- Pour la deuxième méthode « spécification de la structure du système FIS», on suit les étapes suivantes :
 - On ouvre le menu «membership fonctions» à partir du menu «Edit»
 - On ajoute les fonctions d'appartenances désirées pour les entrées et la sortie, (pour la sortie soit une fonction linéaire ou une constante (singleton))
 - On sélectionne le menu Rules du menu «Edit», et on utilise le «Rule Editor» pour générer les règles.
 - On sélectionne le menu «Fis properties» du menu «Edit» pour donner un Nom au système FIS, et pour enregistrer soit dans l'espace de travail ou dans un dossier.
 - On clique sur le bouton «close» pour retourner à l'interface graphique (ANFIS Editor GUI).

✓ **Remarque :**

Pour importer un système FIS déjà généré et sauvegardé, on clique sur «load from file» ou «load from workspace» de la rubrique «generate FIS» de l'interface graphique ANFIS editor pour importer le système FIS initial.

3.5.1.4 Affichage de la structure du système ANFIS

Après la génération du système FIS, on peut afficher le model de la structure du système ANFIS, à l'aide du bouton structure dans le milieu de la partie droite de l'interface graphique (voir Figure 3.9)

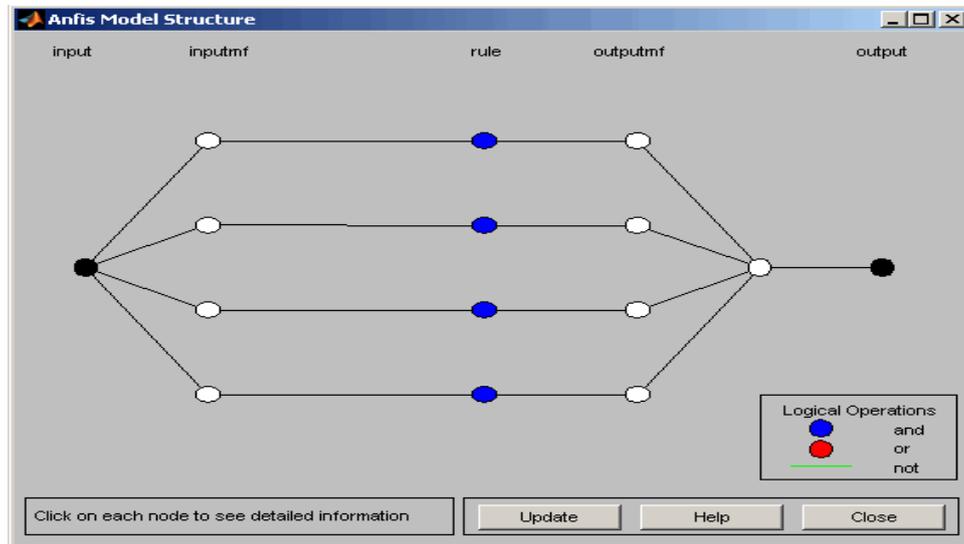


Figure 3.9 structure du système ANFIS

Les branches de cette graphe sont codées par des couleurs, le codage couleur des branches caractérisent les règles, les entrées sont représentées par le nœud le plus à gauche et la sortie par le nœud le plus à droite. On cliquant sur les nœuds on obtient des informations détaillées sur la structure.

3.5.1.5 Apprentissage du système ANFIS

Il y'a deux méthodes d'apprentissage (optimisation des paramètres) disponibles, ces méthodes sont :

- La méthode hybride (hybrid) qui est la méthode par défaut, cette méthode est une méthode mixte, elle utilise la méthode des moindres carrés et la méthode de rétro-propagation.
- La méthode de rétro-propagation.

La tolérance d'erreur est utilisée pour créer un critère d'arrêt d'apprentissage, qui est liée à la taille d'erreur. L'apprentissage s'arrête après que l'erreur de données d'apprentissage reste dans les limites de cette tolérance. Il est préférable de laisser l'erreur tendre vers zéro si nous ne savons pas comment l'erreur d'apprentissage peut se comporter.

Pour démarrer l'apprentissage :

- Sélectionnez la méthode d'optimisation dans la rubrique «train fis».
- Régler le nombre d'itérations et la tolérance d'erreur pour l'apprentissage, dans la rubrique «train fis» de l'interface graphique.
- Lancer l'opération d'apprentissage avec «train now» de la rubrique «train fis», une fenêtre s'affiche (voire Figure 3.10).

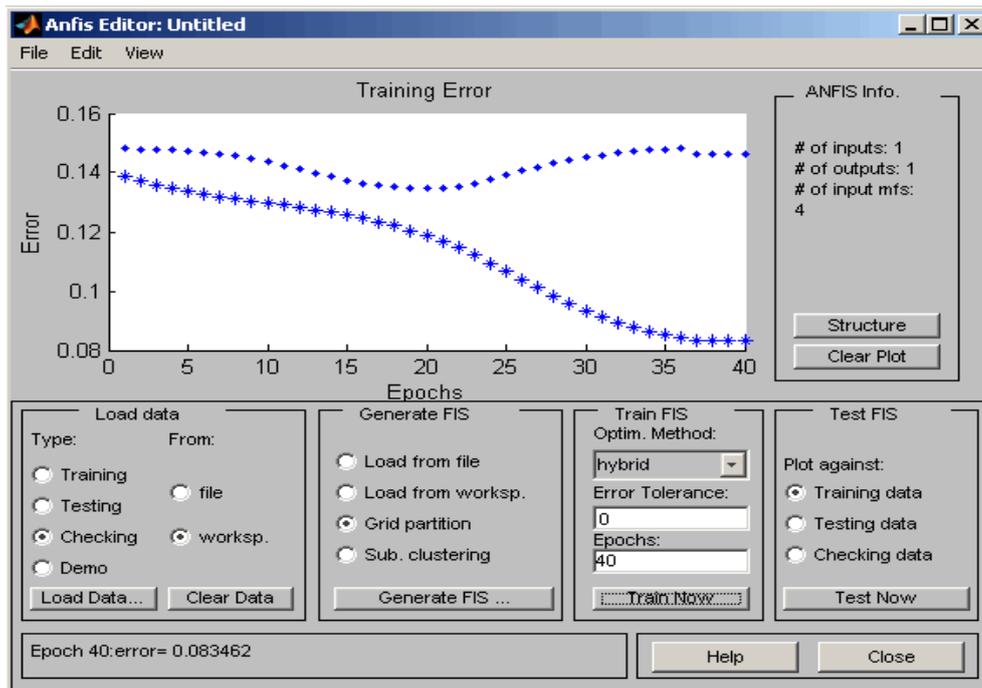


Figure 3.10 apprentissage du système ANFIS

Le graphique montre l'erreur de vérification (checking) comme suit: ♦ ♦ en haut. L'erreur d'apprentissage apparaît sous la forme * * en bas. L'erreur de vérification (checking) diminue jusqu'à un certain point dans l'apprentissage, puis augmente. Cette augmentation représente le point de sur-apprentissage du modèle. ANFIS choisit les paramètres du modèle associés à l'erreur de vérification minimale (juste avant ce point de saut). Cet exemple montre pourquoi l'option de données de vérification pour ANFIS est utile.

3.5.1.6 Validation du système entraîné FIS

Une fois que le système FIS est entraîné, on peut le **valider** en utilisant un jeu de données (**testing data or checking data**) différentes de celles utilisées pour l'apprentissage.

Pour valider le système entraîné FIS:

1. Dans la partie "Load Data" de l'éditeur ANFIS, sélectionnez le jeu de données de validation ("testing data" or "checking data") et cliquez sur "Load Data".
2. Dans la partie "Test FIS", sélectionner le jeu de validation, puis cliquez sur "Test now". Une fenêtre s'affiche (voire la Figure 3.11). Cette fenêtre représente la sortie du système entraîné (FIS output :*) en couleur rouge par rapport à l'entrée (données de validation (checking data :+)) en couleur bleu.

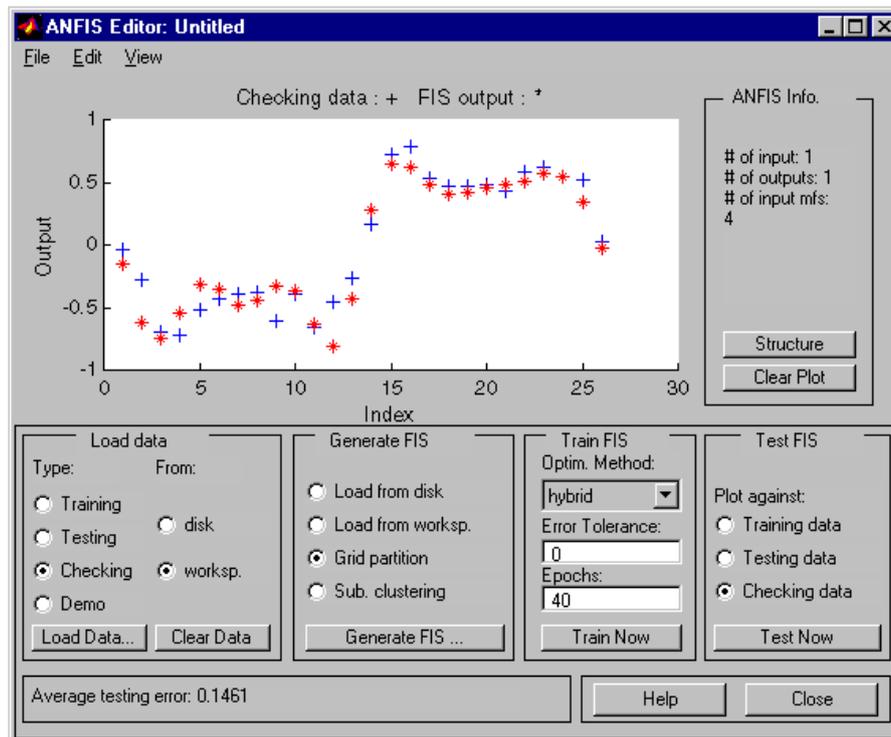


Figure 3.11 test du système ANFIS entraîné

3.6 Description du banc d'essais étudié

3.6.1 Introduction

La base de données étudiée comporte des signaux de roulements provenant du centre de recherche: " *the Case Western Reserve University- Bearing Data Center. Cleveland, Ohio. USA.* ". Les signaux ont été recueillis sur un banc d'essais (pour des roulements à billes normaux et défectueux), composé d'un moteur, un accouplement et une génératrice. Des défauts simples sous forme de points « de différents diamètres et profondeurs » ont été créés sur les différents organes composant les roulements d'essais.

3.6.2 Signaux de roulements "bearing data center"

Ce site Web "www.eecs.cwru.edu/laboratory/bearing/welcome_overview.htm"⁽¹⁾; permet d'accéder à la base de données des essais de roulements à billes, normaux ou défectueux. Cette base de données a été utilisée dans plusieurs études pour valider l'efficacité des nouveaux algorithmes de diagnostic des défauts de roulements (Gryllias et Antoniadis, 2012); (Harrouche et Felkaoui, 2014) ; (Zhang et al, 2013); (Shen et al, 2013) ; (Fedala, 2005) ; (Loparo, 2013) Comme représenté sur la Figure (3.12) ci-dessous, le banc d'essai se compose principalement d'un moteur "2HP"⁽²⁾ (à gauche de l'image), d'un accouplement transducer/encoder (centre), d'un dynamomètre (à droite de l'image) et des circuits de commande (non montrés).

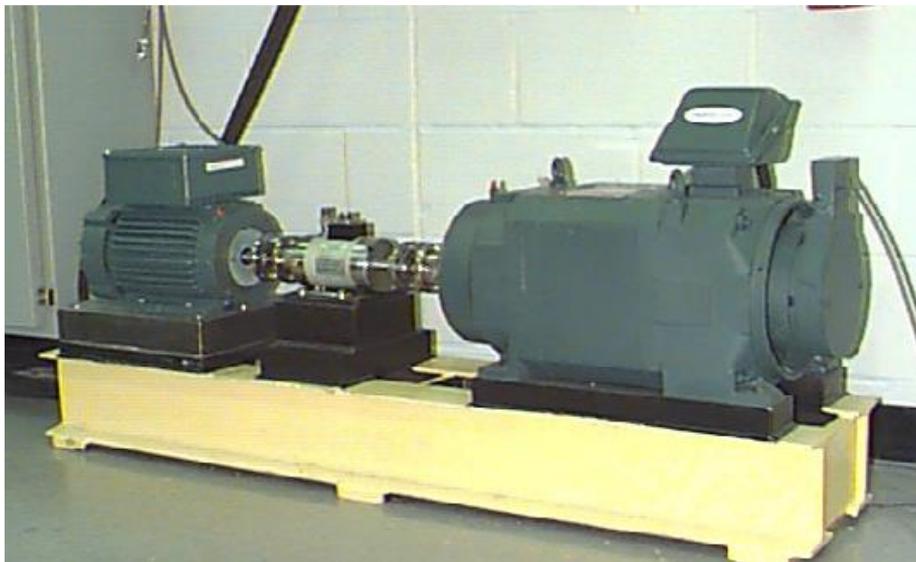


Figure 3. 12 Banc d'essai de roulements (Loparo, 2013).

Les roulements testés soutiennent l'arbre du moteur des deux côtés (coté entraînement et coté ventilateur). De simples défauts sous forme de points ont été créés sur les roulements d'essai, en utilisant l'usinage électro-décharge.

Le Tableau 3.1 Représente les diamètres des défauts, leurs profondeurs et les marques de roulements utilisés

⁽¹⁾ The Case Western Reserve University - Bearing Data Center. Cleveland, Ohio. USA.

⁽²⁾ HP: horsepower, 1hp = 745, 69987 watt

Tableau 3.1 Caractéristiques des défauts de roulements (voir le site "www.eecs.cwru.edu/laboratory/bearing/welcome_overview.htm ")

| Position du roulement | Emplacement du défaut | Diamètre [inches] ¹ | profondeur [inches] | Marque du roulement |
|-----------------------|-----------------------|--------------------------------|---------------------|---------------------|
| coté entraînement | bague Intérieur | 0.007 | 0.011 | SKF |
| | | 0.014 | | |
| | | 0.021 | | |
| | | 0.028 | | |
| | bague extérieur | 0.007 | 0.011 | SKF |
| | | 0.014 | | |
| | | 0.021 | | |
| | | 0.028 | | |
| | Bille | 0.007 | 0.011 | SKF |
| | | 0.014 | | |
| 0.021 | | | | |
| 0.028 | | | | |
| coté ventilateur | bague Intérieur | 0.007 | 0.011 | SKF |
| | | 0.014 | | |
| | | 0.021 | | |
| | bague extérieur | 0.007 | | |
| | | 0.014 | | |
| | | 0.021 | | |
| | Bille | 0.007 | | |
| | | 0.014 | | |
| | | 0.021 | | |
| | | 0.028 | | |

Des roulements **SKF** ont été utilisés pour les défauts de diamètre 0.007, 0.014 et 0.021 [inches⁽¹⁾]. Et des roulements équivalents **NTN** ont été utilisés pour les défauts de diamètre 0.028 et 0.04 [inches].

Les caractéristiques des roulements des deux cotés (accouplement et ventilateur), y compris les fréquences des défauts et la géométrie du roulement sont énumérées ci-dessous: Tableau 3.2.

Tableau 3.2 Caractéristiques des roulements et leurs Fréquences des défauts (voir le site "www.eecs.cwru.edu/laboratory/bearing/welcome_overview.htm ")

| | | | | |
|--|--------------------|-----------------|-------------------|-------------------|
| Caractéristique du roulement coté accouplement:6205-2RS JEM SKF, roulement à Billes de cannelure profonde. | | | | |
| Taille: (inches) | | | | |
| Diamètre Intérieur | Diamètre Extérieur | Épaisseur | Diamètre de Bille | Diamètre primitif |
| 0.9843 | 2.0472 | 0.5906 | 0.3126 | 1.537 |
| Fréquences du défaut : (multiple de vitesse courante en Hz) | | | | |
| | Bague Intérieur | Bague Externe | Cage | Élément Roulant |
| | 5.4152 | 3.5848 | 0.39828 | 4.7135 |
| Caractéristique du roulement coté ventilateur:6203-2RS JEM SKF, roulement à billes de cannelure profonde. | | | | |
| Taille: (inches) | | | | |
| Diamètre Intérieur | Diamètre Extérieur | Épaisseur | Diamètre de bille | Diamètre primitif |
| 0.6693 | 1.5748 | 0.4724 | 0.2656 | 1.122 |
| Fréquences du défaut : (multiple de vitesse courante en Hz) | | | | |
| | Bague Intérieur | Bague extérieur | Cage | Élément Roulant |
| | 4.9469 | 3.0530 | 0.3817 | 3.9874 |

[inches]⁽¹⁾ = pouces=25,4mm

Les signaux ont été enregistrés à l'aide des accéléromètres, qui ont été fixés à la cage du moteur par aimant. Ces capteurs ont été placés à la position 12 heures "vertical", des deux cotés "accouplement et ventilateur" de la cage du moteur.

Les défauts de la bague externe sont des défauts stationnaires, donc le placement du défaut relativement à la zone de charge du roulement à un impact direct sur la réponse vibratoire du système "moteur/roulement". Afin de mesurer cet effet, des essais ont été effectués des deux cotés "accouplement et ventilateur", avec des défauts de bague situés à 3 heures (directement dans la zone de charge), à 6 heures (orthogonale à la zone de charge), et à 12 heures.

Les données ont été enregistrées pour:

- les roulements sans défauts.
- Les défauts seulement sur le coté accouplement (12.000 et 48.000 points/seconde)
- Les défauts seulement sur le coté ventilateur (12.000 points/seconde).

La vitesse [tr.min] et les puissances [HP] pour chaque essai ont été enregistrées manuellement en utilisant le couple transducer/encoder. Tous les fichiers de données sont au format (*.mat) Matlab. Chaque fichier contient trois données: signaux de vibration des deux cotés et la vitesse de rotation du moteur. Pour notre étude ; nous avons considéré seulement les signaux coté accouplement.

La Figure 3.13 représente les signaux temporels recueillis du coté de l'accouplement, pour différents diamètres de défauts, réalisés sur la bague intérieure, pour un couple moteur de 3HP. (HP: Horse Power)

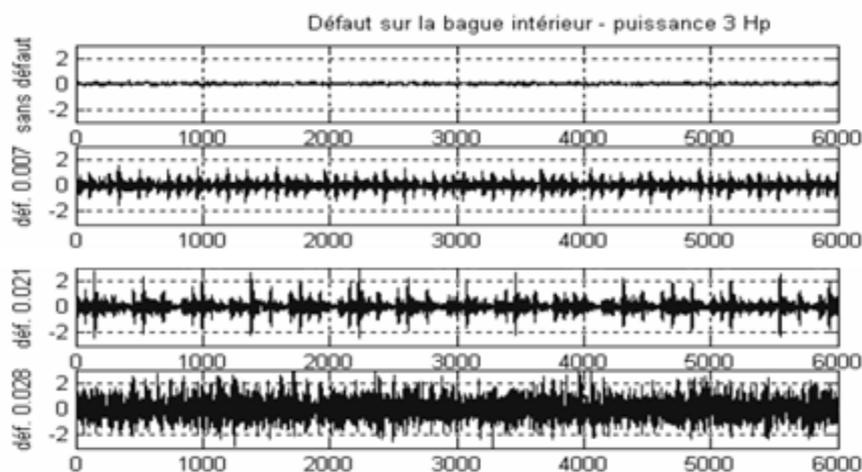


Figure 3.13 Représentation des signaux temporels pour différents diamètres du défaut sur la bague intérieure.

3.7 Prétraitement et extraction des indicateurs du vecteur forme

Pour le prétraitement et l'extraction des paramètres correspondants aux différents états de fonctionnement du système étudié en vue de les utiliser pour la classification; nous avons développé des sous-programmes sous environnement Matlab pour cet objectif (voir [Annexes \(de 4.1 à 4.4, 4.7 et 4.8\)](#), [Organigramme 4.1](#), [Algorithme 4.1](#)).

3.8 Classification des défauts par les deux systèmes FIS et ANFIS

Nous avons développé des programmes sous environnement Matlab, qui nous permet de classifier les différents états de défauts du banc d'essais étudié, en utilisant des systèmes d'inférence floue (FIS) et des systèmes d'inférence neuro-flou adaptés ANFIS(voir [Annexes\(4.5 et 4.6, de 4.9 à 4.13\)](#), [Organigramme 4.2](#)).

3.9 Conclusion

Au cours de ce chapitre, nous avons présenté les étapes les plus importantes que nous suivrons au cours de notre étude, Où nous avons présenté et en détail l'implémentation des deux systèmes FIS et ANFIS sous l'environnement MATLAB, ainsi nous avons également présenté le banc d'essais de roulements et les signaux qui en ont été extraits, ces derniers serviront à la validité des différents systèmes de diagnostic élaborés au cours de notre étude.

Dans ce sens, le chapitre suivant traitera la mise en œuvre de la méthodologie choisie pour notre travail, qui vise à optimiser un système d'inférence floue(FIS) conçu pour le diagnostic des défauts mécaniques des machines tournantes. Cette méthodologie repose sur trois étapes fondamentales. Dans la première étape, nous créons un système d'inférence floue(FIS) dans lequel leurs entrées sont sélectionnées de manière aléatoire à partir de l'ensemble d'indicateurs disponible. Dans la deuxième étape et pour le but d'optimiser le système initial d'inférence floue FIS, nous sélectionnons les meilleurs indicateurs capables de séparer nettement les différentes catégories de défauts à l'aide d'une méthode de sélection automatique, les indicateurs sélectionnés seront utilisés comme entrées pour le nouveau système d'inférence floue FIS optimisé structurellement. Dans la troisième et la dernière étape, nous utilisons un système hybride (ANFIS) pour optimiser le système initial (FIS) paramétriquement. Ce chapitre discutera et interprétera également les différents résultats obtenus.

CHAPITRE 4

RESULTATS ET DISCUSSION

Chapitre 4

Résultats et discussion

4.1 Introduction

Le choix des paramètres du système d'inférence flou (FIS) initial qu'on veut optimiser est basé sur l'expertise de l'utilisateur concernant le problème étudié et le savoir-faire dans l'application de la technique proposée pour leur traitement. La variété et la diversité de ces paramètres et la difficulté dans leurs choix, rendent l'élaboration de ces systèmes un peu difficile, surtout dans le cas où on a un nombre assez important de variables d'entrée du système et par conséquent un nombre important de fonctions d'appartenance associées à ces variables. C'est pour cette raison qu'il faut donner une attention particulière à cette étape, qui est en réalité une étape d'optimisation préliminaire qui concerne la structure du système et qu'on appelle optimisation structurelle. Ensuite, on procèdera à une optimisation qui concerne les paramètres du système (paramètres des fonctions d'appartenance), c'est l'optimisation paramétrique.

Cette partie est consacrée aux résultats et discussion. Nous avons opté dans ce chapitre pour la méthodologie de création d'un système d'inférence flou (FIS) initial, puis on l'a optimisé sur le plan structurel et sur le plan paramétrique afin d'obtenir une meilleure configuration du modèle FIS, ce qui permet d'obtenir de bons résultats dans le diagnostic du système étudié.

L'application de la méthodologie choisie commence par la création d'un système d'inférence floue (FIS) initial avec des paramètres choisis d'une manière arbitraire (au hasard), par la suite on procède à une optimisation structurelle au niveau des variables d'entrée et cela par la sélection des indicateurs les plus pertinents parmi l'ensemble des indicateurs disponibles qu'on utilisera par la suite comme entrées du système FIS. La méthode de sélection automatique utilisée est le «critère de Fisher». Pour une meilleure configuration du système de diagnostic on a appliqué sur ce dernier une seconde optimisation, portant sur les paramètres qui permettra d'ajuster les différentes valeurs des paramètres des fonctions

d'appartenance d'entrée et de sortie, cette deuxième méthode d'optimisation appliquée dans cette méthodologie est une méthode hybride basée sur la logique floue et les réseaux de neurones appelé méthode ANFIS.

La validation pratique des différents modèles de cette méthodologie est effectuée par les signaux vibratoires du banc d'essais de roulements déjà présenté en Chapitre 3 (section 3.6.2).

4.2 Diagnostic des défauts de roulements

Comme avancé en introduction, le déroulement de l'application de la méthodologie choisie pour le diagnostic des défauts de roulements comprend trois étapes qui sont :

- Classification des différents défauts par l'utilisation d'un **système d'inférence floue initial(FIS) de type Sugeno**, avec des paramètres d'entrée choisis arbitrairement (au hasard) parmi l'ensemble des indicateurs extraits à partir des signaux vibratoires du système étudié .
- Dans la deuxième étape, on procède à une **optimisation structurelle** du système initial FIS, en sélectionnant parmi les indicateurs disponibles ceux pertinents par le Critère de Fisher. Ainsi, les indicateurs sélectionnés seront utilisés comme des entrées pour le système FIS. Enfin, nous procéderons à la classification des différents défauts par ce nouveau système FIS optimisé structurellement.
- Pour la troisième étape, une deuxième optimisation est appliquée, c'est l'**optimisation paramétrique**, qui permettra d'ajuster les valeurs des différents paramètres du système d'inférence floue(FIS) initial. La méthode appliquée dans cette étape est une méthode hybride qui combine en même temps les techniques de la logique floue et les réseaux de neurones, c'est la méthode ANFIS. A la fin de cette étape, une opération de classification a été effectuée par le nouveau modèle ANFIS.

Mais, avant de commencer la classification des défauts en suivant les trois étapes, un prétraitement des signaux vibratoires du banc d'essais étudié est nécessaire. Nous décrirons ci-dessous les différentes étapes de prétraitement des signaux vibratoires et la construction de la base de données.

4.2.1 Création de la base de données

L'ensemble des signaux disponibles contient quatre signaux ; un signal sans défauts et les trois autres avec défauts. Comme, les systèmes d'inférence floue(FIS) nécessitent une

grande base de données pour faire la classification (Lou et Loparo, 2004), (Saravanan et al, 2009), (Dawu et al, 2009), (Sugumaran et al, 2007), et pour avoir suffisamment de données, nous avons divisé chaque signal en K_t tranches.

4.2.1.1 Préparation des observations

a) Division des signaux

Pour avoir suffisamment de données, nous avons divisé chaque signal en K_t tranches de N_M points à condition que le nombre d'échantillons pour chaque tranche soit suffisamment grand, afin de couvrir un nombre suffisant du plus grand cycle correspondant à la plus grande période (la plus petite vitesse de rotation).

Nous disposons des signaux de 120800 points. L'arbre tourne avec une vitesse de 1730 rpm donc sa fréquence est :

$$f_{r_{\text{arbre}}} = \frac{1730}{60} = 28.83 \text{HZ}$$

On a quatre fréquences caractéristiques correspondantes aux différents éléments du roulement (cage, bague extérieure et intérieure et les éléments roulants). La plus grande période correspond à la plus petite fréquence.

Dans notre cas, la plus grande période est la période de la cage :

$$f_{\text{cage}} = 0.3982 \times 28.83 = 11.48 \text{HZ}$$

Où ; 0.3982 : Coefficient qui caractérise la cage du roulement (voir tableau 3.2)

28.83 : Fréquence de rotation de l'arbre en (HZ)

- la période de la cage est :

$$T_{\text{cage}} = \frac{1}{f_{\text{cage}}}$$

- la fréquence d'échantillonnage :

$$f_e = 12 \text{KHZ} = 12000 \text{HZ}$$

- la période d'échantillonnage :

$$\Delta t = \frac{1}{f_e}$$

- le nombre de points par période :

$$N_p = \frac{T_{\text{cage}}}{\Delta t} = \frac{f_e}{f_{\text{cage}}} = \frac{12000}{11.48} = 1045.29 \text{pts} \cong 1046 \text{pts.}$$

Si nous prenons un coefficient de 5, la longueur n écessaire pour chaque tranche sera :

$$N_T = N_p \times 5 = 5230 \text{ pts.}$$

Où : 5 : coefficient utilis é pour multiplier le nombre de points constituant la plus grande période (parmi les périodes des différents éléments du roulement), et cela pour que le nombre d'échantillons pour chaque tranche soit suffisamment grand.

Si on prend pour N_M une valeur de 2^n supérieur à N_T

$$N_M = 2^{13} = 8192 \text{ échantillons.}$$

Alors le nombre de tranches dans le cas d'un recouvrement de 50%, soit $s_1 = \frac{N_M}{2}$

Alors :

$$K_t = \frac{N_s - s_1}{N_M - s_1} = \frac{120800 - 4096}{8192 - 4096} \cong 28 \text{ Tranches.}$$

Donc, chaque signal est divis é en 28 tranches.

Pour les quatre signaux, nous avons obtenu 112 tranches ; 28 tranches (signaux) pour chaque type de défaut ou pour chaque classe.

La Figure 4.1 présente le découpage d'un signal en tranches ou sections.

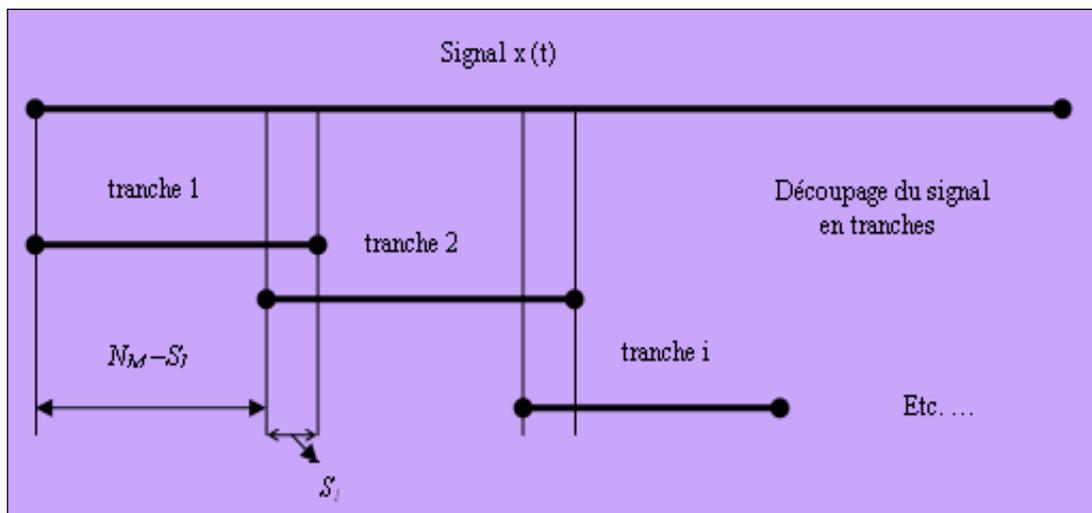


Figure 4.1 Découpage d'un signal en tranches ou sections

b) Pr étraitement et calcul des indicateurs

Nous avons choisi initialement de calculer les indicateurs suivants (Harrouche et Felkaoui, 2014) : l'écart_type(ET), le facteur de crête (F_c), la valeur crête-crête (A_{cc}), le

facteur de dissymétrie (skewness en anglais(K_s)) et le facteur d'aplatissement (Kurtosis(K_u)), ainsi que les énergies issues du spectre d'enveloppe. Ces indicateurs peuvent être de bons indicateurs de surveillance dans le cas des roulements (Breneur, 2002), (Ping, 2000), (Sun, 1998).

Après une analyse préliminaire, nous avons choisi de calculer ces indicateurs comme suit :

b.1) Indicateurs temporels

Les indicateurs temporels cités précédemment sont calculés dans 5 bandes fréquentielles issues de la bande d'analyse '6KHZ', 4 sous bandes adjacentes de largeur 1500HZ, plus la bande totale. Les indicateurs considérés sont les moyennes des indicateurs calculés dans les cinq bandes. Les bandes de calcul sont alors : [0-1500HZ], [1500-3000HZ], [3000-4500HZ], [4500-6000HZ] et [0-6000HZ]. Le signal de chaque tranche a donc été centré puis filtré par des filtres de type passe-bande.

b.2) Indicateurs fréquentiels

Les indicateurs fréquentiels sont des énergies issues du spectre d'enveloppe des différentes tranches calculés dans la bande totale et en 5 bandes fréquentielles de largeur de 1000HZ. Les indicateurs fréquentiels sont : E_{BT} (énergie calculée dans la bande totale [0-6000HZ]), E_{B1} (énergie dans la bande 1 [0-1000HZ]), E_{B2} (énergie dans la bande 2 [1000-2000HZ]), E_{B3} (énergie dans la bande 3 [2000-3000HZ]), E_{B4} (énergie dans la bande 4 [3000-4000HZ]) et E_{B5} (énergie dans la bande 5 [4000-5000HZ]).

- Remarques

- ✓ A la fin de cette phase de préparation des observations, on peut visualiser la division des 4 signaux en 28 tranches. L'exécution d'un sous-programme, sous environnement Matlab (voir annexe 4.1) conçu pour cet objectif, nous donne la division en tranches des 4 signaux. Les Figures 4.2 et 4.3 présentent respectivement la division du signal (sans défaut), et du signal avec défaut de diamètre de (0.028)inch .
- ✓ On remarque sur les deux Figures qui suivent une distinction claire entre les comportements vibratoires des tranches du signal (sans défaut) et les tranches du signal avec défaut (0,028inch). Cette distinction est due au défaut créé sur la bague intérieure du roulement (0,028inch), Ce défaut entraîne une augmentation sur les deux niveaux vibratoire et énergétique pour les vibrations associés à ce dernier.

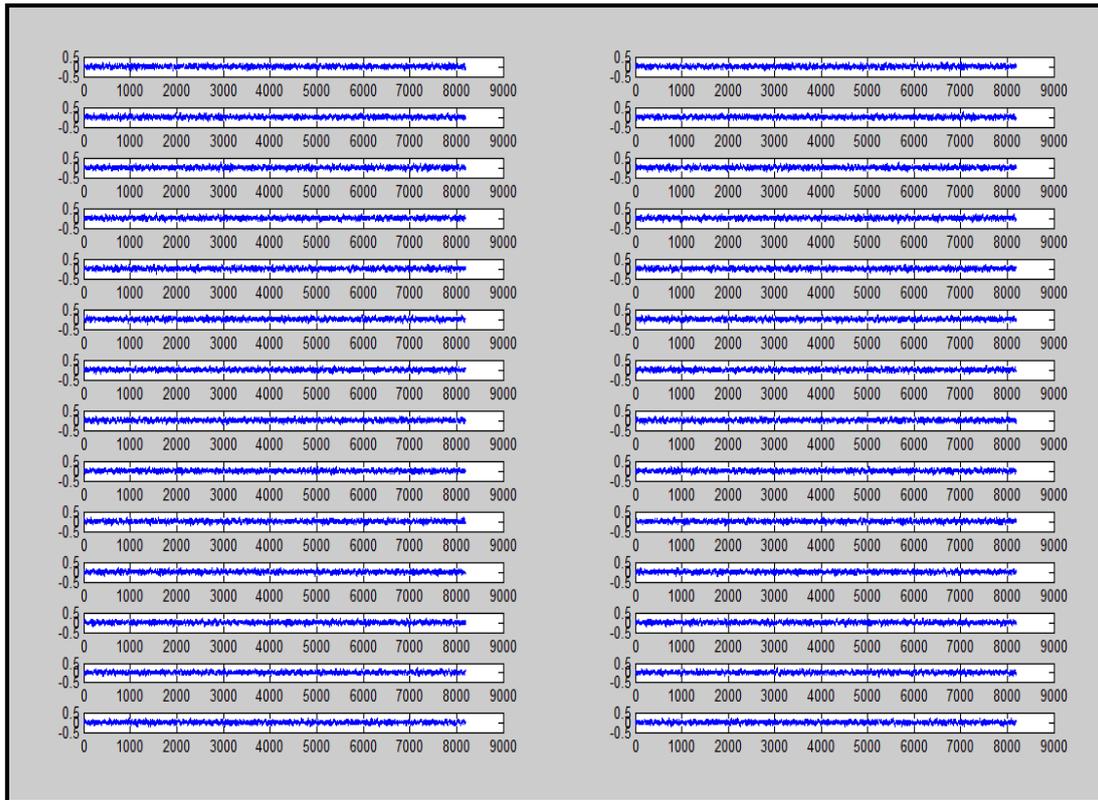


Figure 4.2 Division en tranches du signal sans défaut.

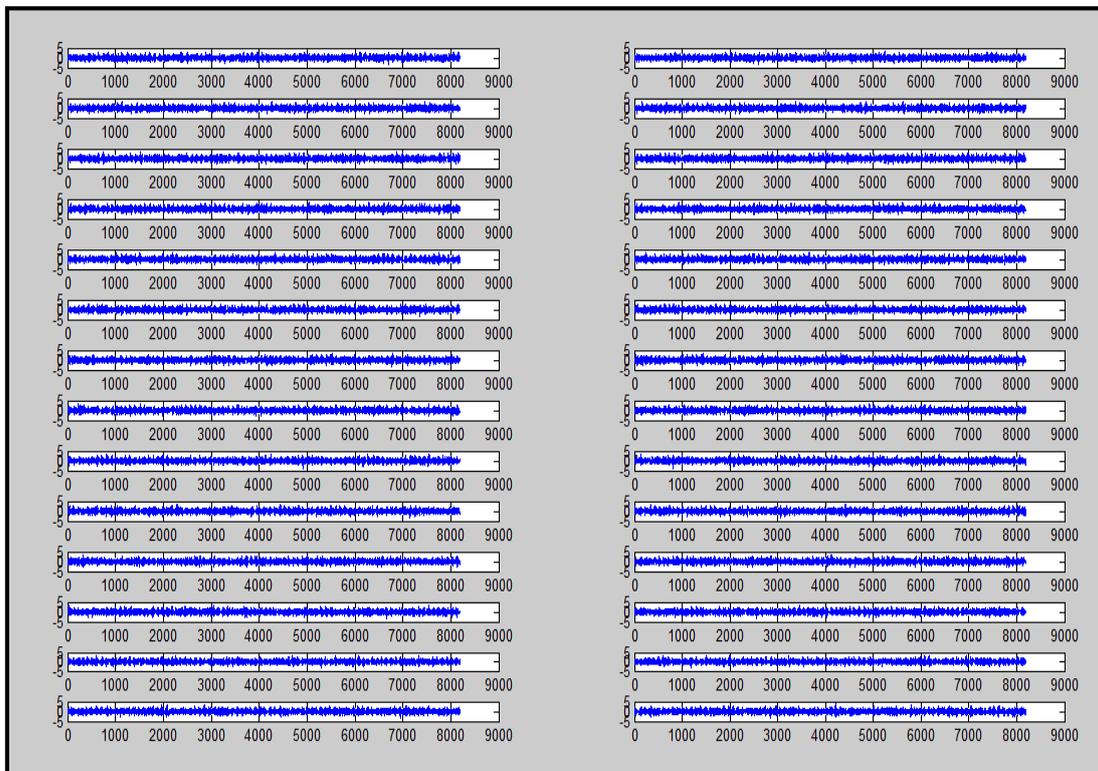
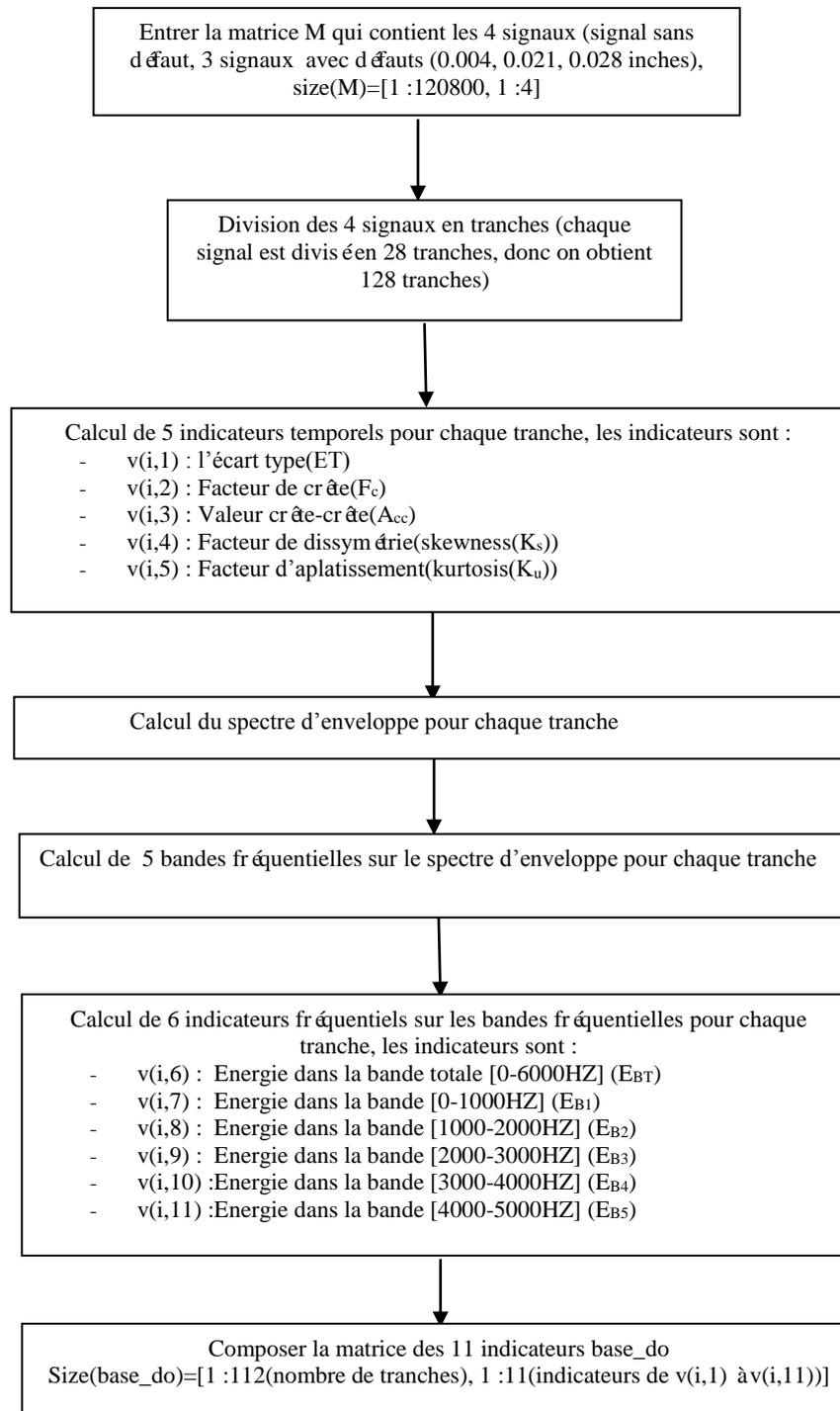


Figure 4.3 Division en tranches du signal avec défaut de diamètre 0,028inch.

- ✓ pour le calcul des indicateurs temporels et fréquentiels (voir **Organigramme 4.1**), un programme sous environnement Matlab est conçu pour cet objectif (voir **annexe (4.2)**).



Organigramme 4.1 Calcul des 11 indicateurs temporels et fréquentiels

- ✓ les Tableaux 4.1 et 4.2 (voir **annexe 4.3**) montrent respectivement la matrice des 11 indicateurs pour les 28 tranches du signal sans défaut et signal avec défaut (diamètre

(0.007inch)), les indicateurs de 1 à 11 sont respectivement : ET, F_c , A_{cc} , K_s , K_u , E_{BT} , E_{B1} , E_{B2} , E_{B3} , E_{B4} , E_{B5} .

✓ les Figures 4.4 à 4.14 représentent graphiquement la variation des 11 indicateurs en fonction des 112 observations (nombre de tranches total).

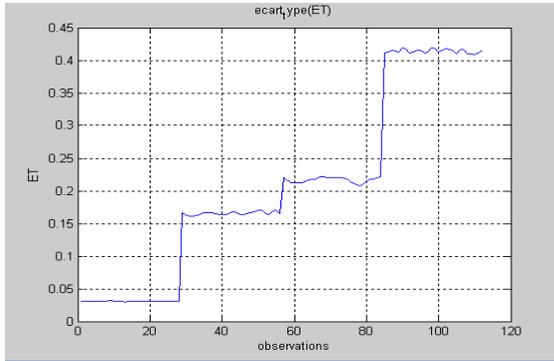


Fig 4.4 Variation de l'indicateur(ET).

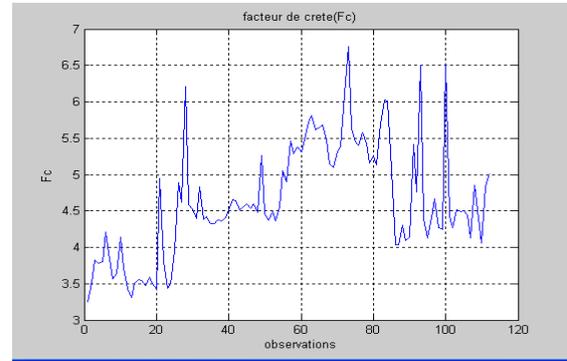


Fig 4.5 Variation de l'indicateur(Fc).

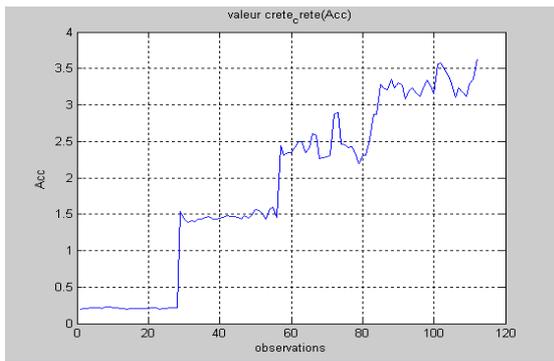


Fig 4.6 Variation de l'indicateur(Acc).

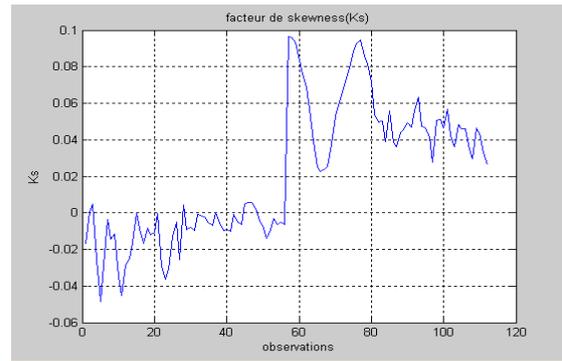


Fig 4.7 Variation de l'indicateur(Ks).

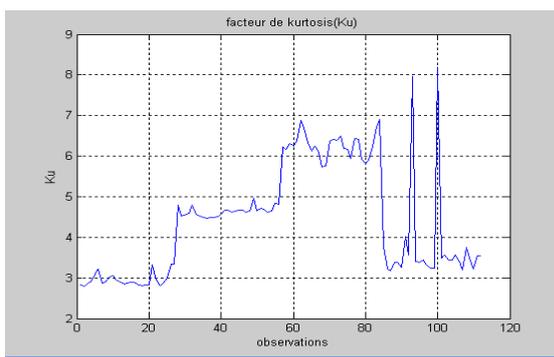


Fig 4.8 Variation de l'indicateur(Ku).

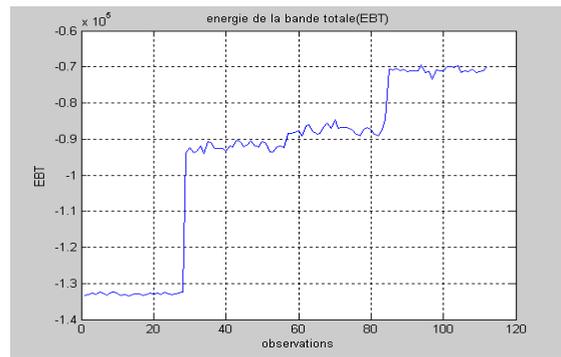


Fig 4.9 Variation de l'indicateur(EBT).

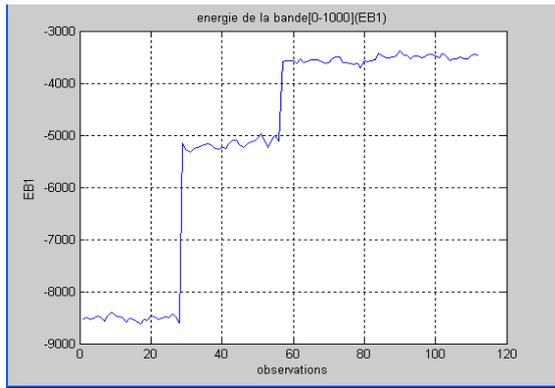


Fig 4.10 Variation de l'indicateur(EB1).

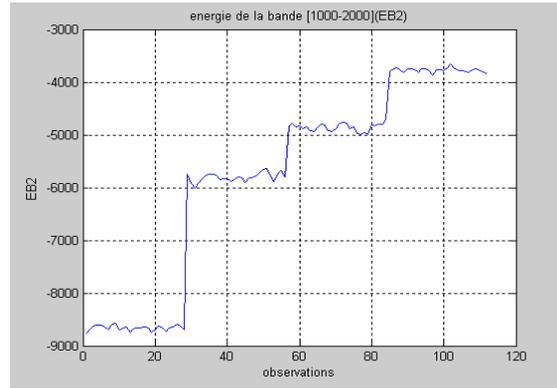


Fig 4.11 Variation de l'indicateur(EB2).

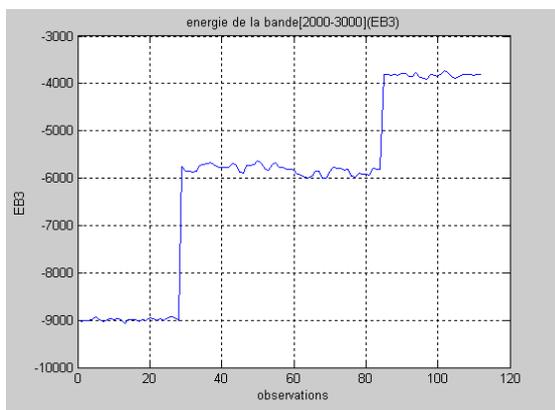


Fig 4.12 Variation de l'indicateur(EB3).

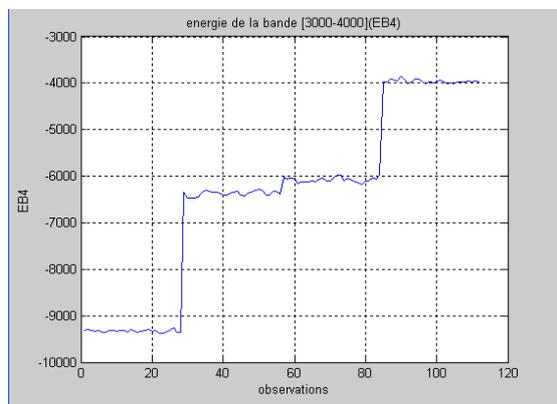


Fig 4.13 Variation de l'indicateur(EB4).



Fig 4.14 Variation de l'indicateur(EB5).

4. 2.1.2 Constitution du vecteur forme de la base de données (vecteur d'indicateurs)

La composante du vecteur forme est constituée par les indicateurs calculés précédemment. Les données que l'on doit classer et traiter sont stockées dans un tableau de type observations/variables de la forme suivante :

$$base - donn\acute{e}es = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1M} \\ x_{21} & x_{22} & & x_{2M} \\ \vdots & & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{NM} \end{pmatrix} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix}$$

Avec : N : le nombre d'observations

M : le nombre d'indicateurs

On a alors :

N observations représentées en ligne qui seront divisées en N_c classes (w_1, w_2, \dots, w_{N_c}) ou modes de fonctionnement, Chaque classe w_i possède N_i observations.

Pour notre travail :

- N=112 ; observations (28 tranches pour chaque signal parmi les 4 signaux).
- M=11 ; le nombre d'indicateurs calculé dans la section (4.2.1.1.b).

4.2.2 Choix des classes

Puisque nous sommes dans le cadre d'une classification supervisée, nous devons choisir les différentes classes que nous utiliserons ultérieurement dans la classification, et (ca) ce en fonction des modes de fonctionnement connus du système étudié et qui sont associés aux différents diamètres de défauts. Nous avons choisi de diviser la base de données en quatre classes, comme indiqué dans le tableau (4.3) ci-dessous.

Tableau 4.3 – Répartition de la base de données en classes.

| Classes (Niveau de sévère du défaut) | Diamètre du défaut (inch) |
|--------------------------------------|---------------------------|
| N1 | Sans défaut |
| N2 | 0.007 |
| N3 | 0.021 |
| N4 | 0.028 |

Donc la répartition de la base de données en classes est la suivante :

- De la 1^{ère} ligne à la ligne 28: classe1 ou niveau1 (N1) (sans défaut).
- De la 29^{ème} ligne à la ligne 56: classe2 ou niveau2 (N2) (avec défaut ($\Phi= 0.007$ inch)).
- De la 57^{ème} ligne à la ligne 84: classe3 ou niveau3 (N3) (avec défaut ($\Phi= 0.021$ inch)).
- De la 85^{ème} ligne à la ligne 112: classe4 ou niveau4 (N4) (avec défaut ($\Phi= 0.028$ inch)).

4.2.3 Classification des défauts de roulements

Comme nous l'avons déjà décrit dans la méthodologie adoptée pour cette étude (section 4.2), nous allons construire dans la première étape de classification le système initial (FIS) avec des paramètres d'entrées choisies arbitrairement parmi l'ensemble d'indicateurs disponible, puis dans les deuxième et troisième étapes de classification, nous optimisons le système FIS de deux manières différentes. La première est liée à sa structure (**optimisation structurelle**) et la deuxième liée à ces paramètres (**optimisation paramétrique**).

4.2.3.1 Classification des défauts de roulements avec un système (FIS) non optimisé (avec des variables d'entrée choisies arbitrairement)

Le système de diagnostic utilisé dans cette première étape, est un système d'inférence floue (SIF) de type Sugeno. Les signaux étudiés ont été enregistrés pour les défauts de la bague intérieure dans les conditions suivantes (voir paragraphe 3.6.2) :

- l'arbre tourne à une vitesse de 1730tr/min.
 - la puissance du couple résistant est de 3HP.
 - Les quatre signaux étudiés sont prélevés du côté accouplement, un signal sans défauts, et les trois autres avec défauts sur la bague intérieure (trois défauts de diamètres 0.007, 0.021, et 0.028 inches. Ces défauts sont sous forme de points, et qui ont été créés sur les roulements d'essais (un seul défaut par roulement), en utilisant l'usinage électro-décharge.).
 - Pour les variables d'entrée du système, nous avons sélectionné de manière aléatoire trois indicateurs parmi les indicateurs calculés précédemment à savoir, le facteur de crête (F_c), le facteur de dissymétrie (skewness en anglais (K_s)), et le facteur d'aplatissement (Kurtosis (K_u)). Nous n'avons choisi que trois indicateurs parce que plus le nombre de variables d'entrée et le nombre de fonctions d'appartenance associés à ces entrées sont importants, plus la génération des règles d'inférence floue devient plus difficile et plus complexe. Par exemple, pour deux entrées et 4 fonctions d'appartenance pour chacune des deux entrées, on aura ($4^2 = 16$) règles d'inférence floue. Dans le cas de trois entrées et 4 fonctions d'appartenance pour chacune des trois entrées, on aura ($4^3 = 64$) règles.
- ✓ **remarque** : l'annexe 4.4 présente un tableau qui contient la base de données selon les conditions citées dans ce paragraphe. Les indicateurs de 1 à 3 sont respectivement : F_c , K_s , K_u .

4.2.3.1.1 Implémentation du système d'inférence flou(FIS) initial sous Matlab

L'implémentation du système d'inférence flou sous Matlab, peut être réalisée par deux méthodes différentes, soit à l'aide des commandes de la boîte à outils ou par l'interface graphique de cette dernière.

a) Implémentation du système (FIS) par interface graphique

a.1) Structure du système d'inférence flou(FIS) initial

Nous avons choisi, un système d'inférence flou de type sugeno avec trois variables à l'entrée (F_c , K_s , K_u) et une seule variable à la sortie qui représente l'état de santé du système étudié (**état du système**). Un mécanisme d'inférence flou fonctionnant avec les méthodes listées ci-dessous a été choisi. Ces méthodes apparaissent aussi sur l'interface graphique montrée sur la figure 4.15. En sortie, une seule variable est produite indiquée par « état ».

- Nom_système : **diag_roul_sugeno2**,
- Type : **Sugeno**,
- ET_méthode : méthode utilisée pour l'opérateur ET (**prod**),
- OU_méthode : méthode utilisée pour l'opérateur OU (**probor**),
- Defuzz_méthode : méthode de défuzzification (**Wtaver**),

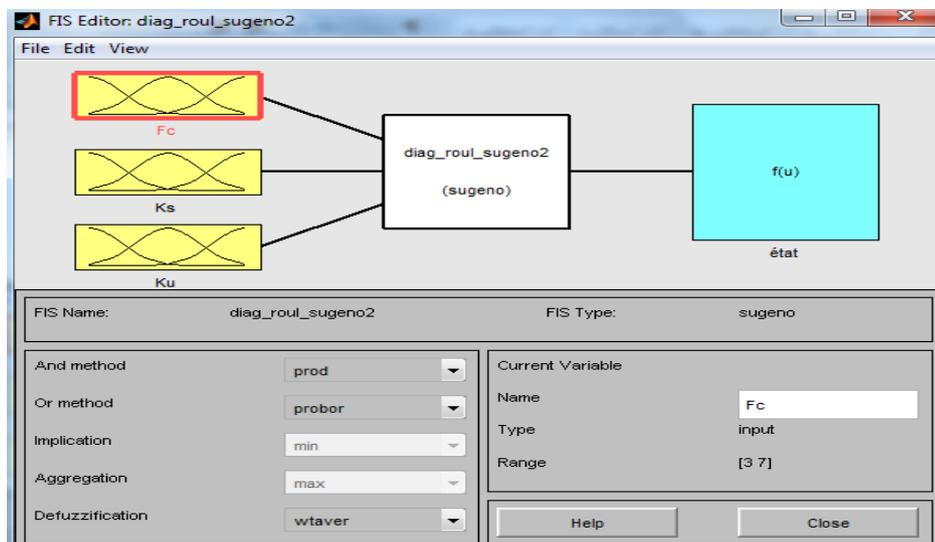


Figure 4.15 Schéma bloc du (SIF).

a.2) Fuzzification des variables d'entrée et de sortie

a.2.1) Fuzzification de la variable d'entrée «le facteur de crête (F_c)»

Cette variable varie dans l'intervalle [$\min(F_c) = 3.2566$, $\max(F_c) = 6.7603$] (voir [annexe \(4.4\)](#)). Donc on a choisi un domaine de variation de [3.0000 7.0000], et quatre fonctions d'appartenance de type gaussienne (une fonction gaussienne se caractérise par deux

paramètres $[\sigma, \mu]$ où σ est l'écart type et μ est l'espérance mathématique). La première fonction appelé «très petite(TP) » avec deux paramètres $[\sigma =0.6347, \mu= 3.8569]$, la deuxième fonction appelé «petite(P) » avec deux paramètres $[\sigma= 0.2218, \mu= 4.5545]$, la troisième fonction appelé « grande(G) » avec deux paramètres $[\sigma= 0.3579, \mu= 5.5517]$, et la dernière fonction appelé «très grande(TG)» avec deux paramètres $[\sigma =0.6393, \mu= 4.6049]$. La Figure 4.16 montre la fuzzification de la variable d'entrée « le facteur de crête (F_c) ».

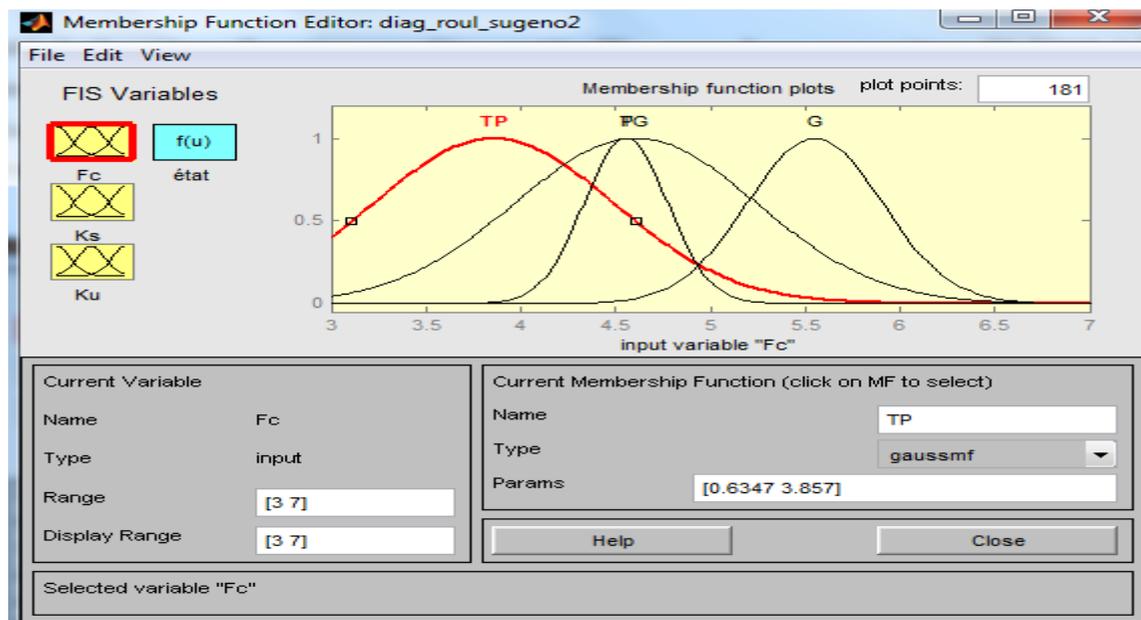


Figure 4.16 Fuzzification de la variable d'entrée « le facteur de crête (F_c) ».

a.2.2) Fuzzification de la variable d'entrée «facteur de dissymétrie (skewness (K_s))»

Cette variable varie dans l'intervalle $[\min(K_s)= -0.0486, \max(K_s)= 0.0963]$ (voir [annexe \(4.4\)](#)). Donc on a choisi un domaine de variation de $[-0.1000 \ 0.1000]$, et quatre fonctions d'appartenance de type gaussienne. La première fonction « très petite(TP) » avec deux paramètres $[\sigma =0.0145, \mu= -0.0174]$, la deuxième fonction appelé «petite(P) » avec deux paramètres $[\sigma =0.0051, \mu= -0.0045]$, la troisième fonction appelé « grande(G) » avec deux paramètres $[\sigma =0.0243, \mu= 0.0637]$, et la dernière fonction appelé «très grande(TG)» avec deux paramètres $[\sigma =0.0089, \mu= 0.0442]$. La Figure 4.17 montre la fuzzification de la variable d'entrée «facteur de dissymétrie (skewness (K_s))».

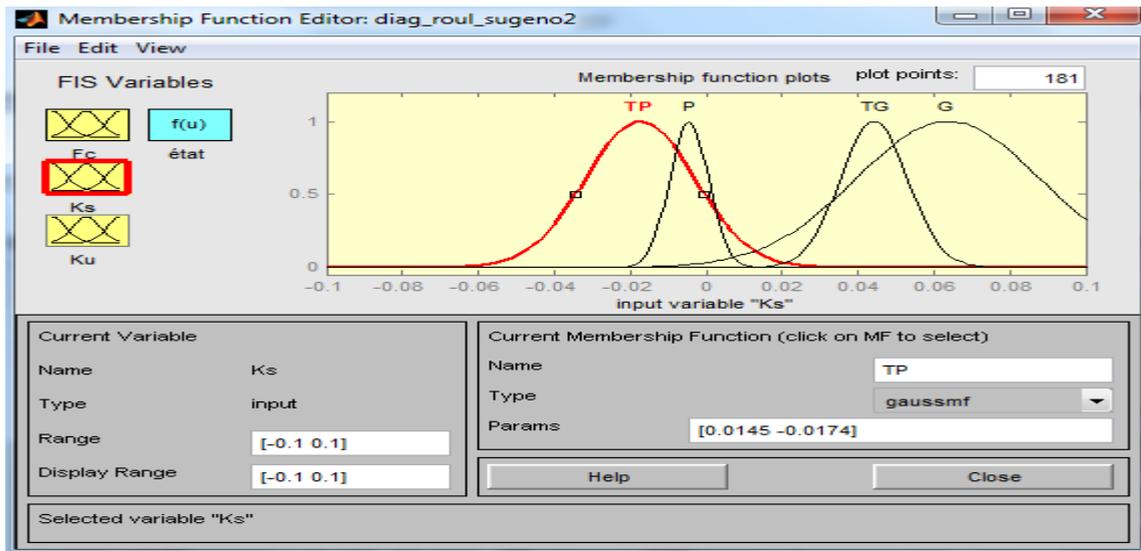


Figure 4.17 Fuzzification de la variable d’entrée «le facteur de dissymétrie (skewness (K_s))»

a.2.3) Fuzzification de la variable d’entrée «facteur d’aplatissement (Kurtosis(K_u))»

Cette variable varie dans l’intervalle [$\min(K_u)= 2.7847$, $\max(K_u)= 8.1944$] (voir annexe (4.4)). Donc on a choisi un domaine de variation de [2.0000 9.0000], et quatre fonctions d’appartenance de type gaussienne. La première fonction appelée «très petite(TP)» avec deux paramètres [$\sigma= 0.3833$, $\mu= 3.0207$], la deuxième fonction appelé «petite(P)» avec deux paramètres [$\sigma= 0.1166$, $\mu= 4.6313$], la troisième fonction appelé «grande(G)» avec deux paramètres [$\sigma= 0.3053$, $\mu= 6.2682$], et la dernière fonction appelé «très grande(TG)» avec deux paramètres [$\sigma= 1.2313$, $\mu= 3.7710$]. La Figure 4.18 montre la fuzzification de la variable d’entrée «facteur d’aplatissement (Kurtosis(K_u))»

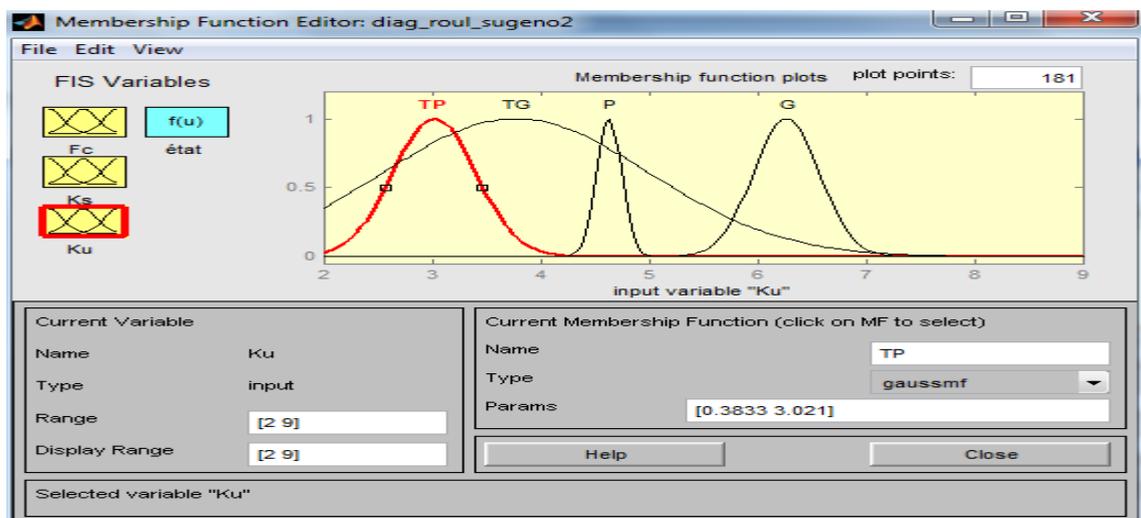


Figure 4.18 Fuzzification de la variable d’entrée «facteur d’aplatissement (Kurtosis(K_u))»

a.2.4) Fuzzification de la variable de sortie « état du dispositif (état) »

On a choisi pour cette variable de sortie un domaine de variation $[0, 10]$, et comme il ya 4 ensembles flous pour chacune des trois variables d'entrée, la variable de sortie sera définie sur 64 ensembles flous. Les fonctions d'appartenances représentant les différents ensembles flous de la variable de sortie sont toutes de type (constante). La Figure 4.19 montre la fuzzification de la variable de sortie « état du dispositif (état) ».

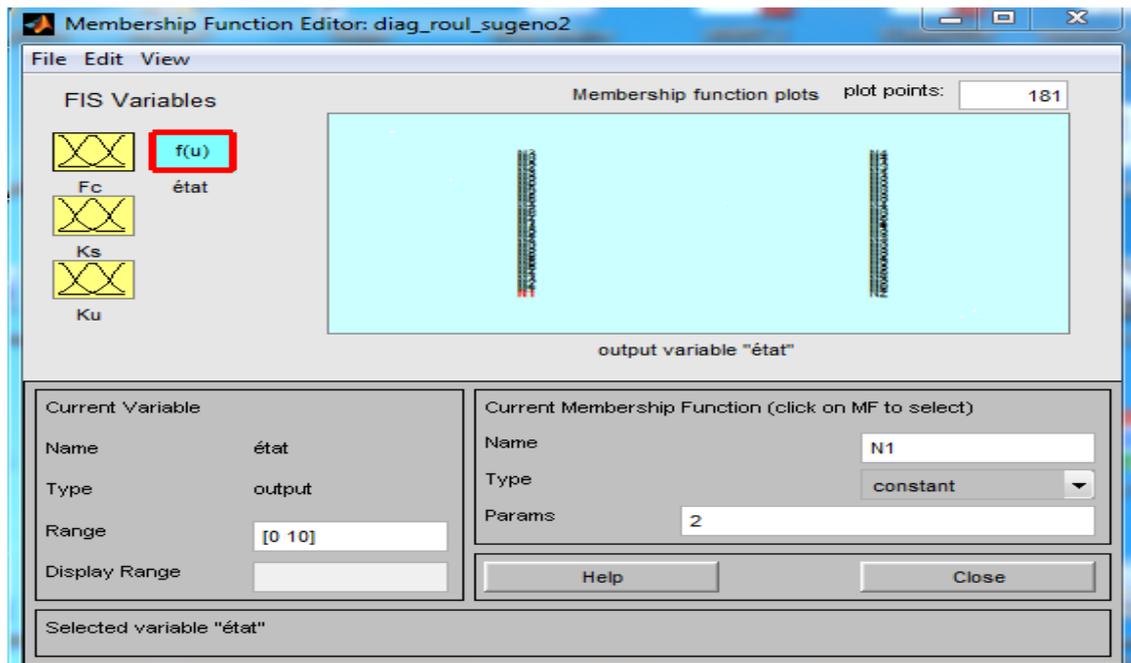


Figure 4.19 Fuzzification de la variable de sortie « état du dispositif (état) ».

a.3) Génération des règles d'inférence

Les différentes règles possibles selon la fuzzification des différentes variables d'entrée et de sortie choisis sont les suivantes (3 variables d'entrée et 4 fonctions d'appartenances pour chacune des variables, on obtient 64 règles d'inférences ($4^3=64$)):

1. si Fc est TP et Ks est TP et Ku est TP alors état est N1
2. si Fc est TP et Ks est TP et Ku est P alors état est N1
3. si Fc est TP et Ks est TP et Ku est G alors état est N2
4. si Fc est TP et Ks est TP et Ku est TG alors état est N3
5. si Fc est TP et Ks est P et Ku est TP alors état est N1
6. si Fc est TP et Ks est P et Ku est P alors état est N2
7. si Fc est TP et Ks est P et Ku est G alors état est N2

8. si Fc est TP et Ks est P et Ku est TG alors état est N2
9. si Fc est TP et Ks est G et Ku est TP alors état est N2
10. si Fc est TP et Ks est G et Ku est P alors état est N2
11. si Fc est TP et Ks est G et Ku est G alors état est N3
12. si Fc est TP et Ks est G et Ku est TG alors état est N3
13. si Fc est TP et Ks est TG et Ku est TP alors état est N2
14. si Fc est TP et Ks est TG et Ku est P alors état est N2
15. si Fc est TP et Ks est TG et Ku est G alors état est N3
16. si Fc est TP et Ks est TG et Ku est TG alors état est N3
17. si Fc est P et Ks est TP et Ku est TP alors état est N1
18. si Fc est P et Ks est TP et Ku est P alors état est N2
19. si Fc est P et Ks est TP et Ku est G alors état est N2
20. si Fc est P et Ks est TP et Ku est TG alors état est N2
21. si Fc est P et Ks est P et Ku est TP alors état est N2
22. si Fc est P et Ks est P et Ku est P alors état est N2
23. si Fc est P et Ks est P et Ku est G alors état est N2
24. si Fc est P et Ks est P et Ku est TG alors état est N3
25. si Fc est P et Ks est G et Ku est TP alors état est N2
26. si Fc est P et Ks est G et Ku est P alors état est N2
27. si Fc est P et Ks est G et Ku est G alors état est N3
28. si Fc est P et Ks est G et Ku est TG alors état est N3
29. si Fc est P et Ks est TG et Ku est TP alors état est N2
30. si Fc est P et Ks est TG et Ku est P alors état est N3
31. si Fc est P et Ks est TG et Ku est G alors état est N3
32. si Fc est P et Ks est TG et Ku est TG alors état est N3
33. si Fc est G et Ks est TP et Ku est TP alors état est N2
34. si Fc est G et Ks est TP et Ku est P alors état est N2
35. si Fc est G et Ks est TP et Ku est G alors état est N2
36. si Fc est G et Ks est TP et Ku est TG alors état est N3
37. si Fc est G et Ks est P et Ku est TP alors état est N2
38. si Fc est G et Ks est P et Ku est P alors état est N2
39. si Fc est G et Ks est P et Ku est G alors état est N3
40. si Fc est G et Ks est P et Ku est TG alors état est N3
41. si Fc est G et Ks est G et Ku est TP alors état est N2

42. si Fc est G et Ks est G et Ku est P alors état est N3
43. si Fc est G et Ks est G et Ku est G alors état est N3
44. si Fc est G et Ks est G et Ku est TG alors état est N3
45. si Fc est G et Ks est TG et Ku est TP alors état est N3
46. si Fc est G et Ks est TG et Ku est P alors état est N3
47. si Fc est G et Ks est TG et Ku est G alors état est N3
48. si Fc est G et Ks est TG et Ku est TG alors état est N4
49. si Fc est TG et Ks est TP et Ku est TP alors état est N2
50. si Fc est TG et Ks est TP et Ku est P alors état est N2
51. si Fc est TG et Ks est TP et Ku est G alors état est N3
52. si Fc est TG et Ks est TP et Ku est TG alors état est N3
53. si Fc est TG et Ks est P et Ku est TP alors état est N2
54. si Fc est TG et Ks est P et Ku est P alors état est N3
55. si Fc est TG et Ks est P et Ku est G alors état est N3
56. si Fc est TG et Ks est P et Ku est TG alors état est N3
57. si Fc est TG et Ks est G et Ku est TP alors état est N3
58. si Fc est TG et Ks est G et Ku est P alors état est N3
59. si Fc est TG et Ks est G et Ku est G alors état est N3
60. si Fc est TG et Ks est G et Ku est TG alors état est N4
61. si Fc est TG et Ks est TG et Ku est TP alors état est N3
62. si Fc est TG et Ks est TG et Ku est P alors état est N3
63. si Fc est TG et Ks est TG et Ku est G alors état est N4
64. si Fc est TG et Ks est TG et Ku est TG alors état est N4

a.4) Défuzzification de la variable de sortie

✓ Exemple d'application

On prend un exemple d'application pour une observation de la première classe (sans défaut(N1)) et on le classe ensuite par le système FIS :

$X = [3.2566 \ -0.0167 \ 2.8269]$ (voir la première ligne de la matrice de [l'annexe \(4.4\)](#)).

La sortie réelle du système FIS pour le vecteur X est : état=3.95≈4

On remarque que notre système FIS a mal classé cette observation (état=3.95≈4), elle est classée dans la deuxième classe (N2) au lieu de la classe (N1).

Ces valeurs sont portées à titre d'exemple, le système proposé FIS accepte toutes les observations de la base de données. La Figure 4.20 montre la défuzzification pour l'exemple pratique donné

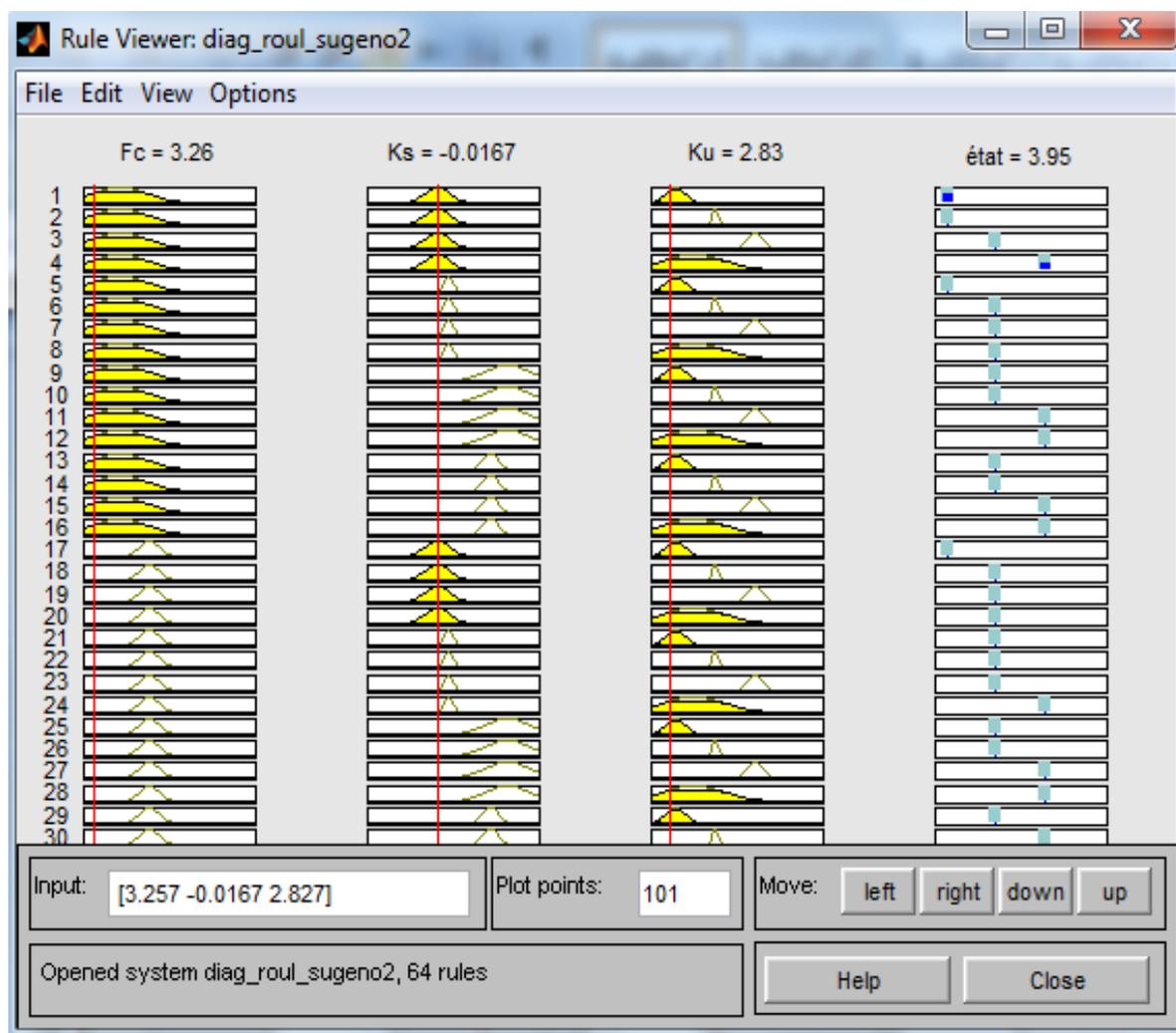
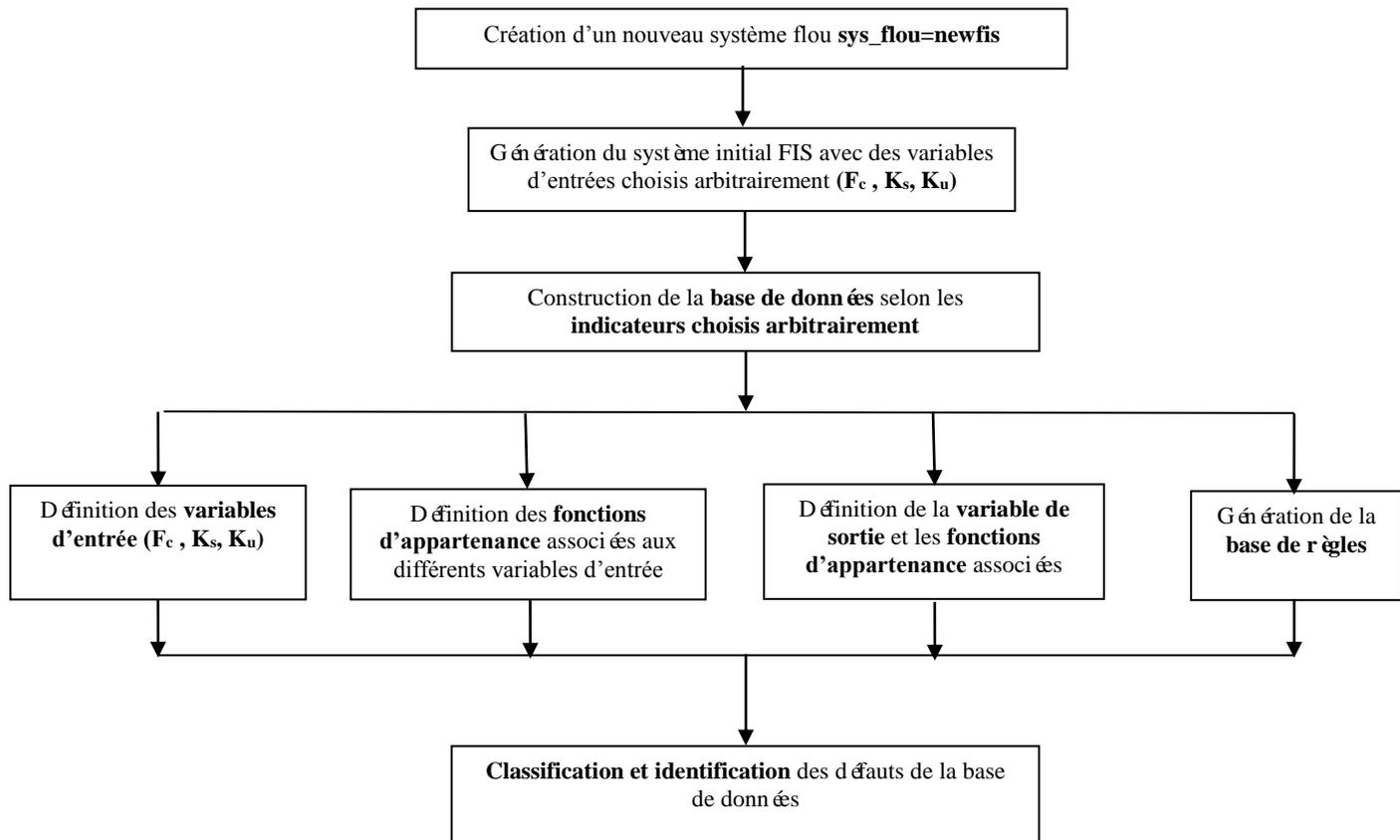


Figure 4.20 Défuzzification de la variable de sortie (état).

b) Implémentation du système(FIS) initial par les commandes de la boîte à outils «Fuzzy logic»

Nous avons conçu un programme sous environnement Matlab(voir [Organigramme 4.2](#)) à l'aide des commandes de la boîte à outils «fuzzy logic » et cela pour l'implémentation du système d'inférence floue initial FIS (avant l'optimisation) et aussi pour la classification des défauts de roulements (le déroulement de l'implémentation est déjà présenté dans la section (3.4) du chapitre 3. Le programme développé est présent dans [l'annexe 4.5](#).



Organigramme 4.2 Implémentation du système FIS initial sous environnement MATLAB par les commandes de la boîte à outils «fuzzy logic »

4.2.3.1.2 Classification de la base de données par le système (FIS) initial (avant l'optimisation)

Le programme développé dans la section 4.2.3.1.1.b, permet de classer les observations de la base de données. L'annexe (4.6) présente le résultat de classification de la base de données.

Dans l'annexe (4.6), on a :

En ligne: une seule ligne ; qui représente la variable de sortie.

En colonne : 112 colonnes ; portent les valeurs réelles de la variable de sortie (état du dispositif étudié (état)).

a) Discussion des résultats de classification

a.1) Représentation graphique du résultat de classification

La Figure 4.21.a représente graphiquement le résultat de classification, et la Figure 4.21.b montre l'erreur absolue de la sortie du système FIS initial par rapport à la sortie souhaitée (désirée), une observation est considérée comme mal classée lorsque l'erreur absolue pour cette dernière est supérieure ou égale à 1.

On remarque que :

- les observations considérées au départ de la classe1 (sans défauts) qui correspondent aux lignes 1 à 28 dans la matrice des données, 28 observations de cette classe sur 28 sont mal classées par le système (FIS) initial.
- les observations considérées au départ de la classe2 (diamètre du défaut (0.007inch)) qui correspondent aux lignes 29 à 56 dans la matrice des données, 24 observations de cette classe parmi 28 sont bien classées par le système(FIS) initial.
- les observations considérées au départ de la classe3 (diamètre du défaut (0.021inch)) qui correspondent aux lignes 57 à 84 dans la matrice des données, 27 observations de cette classe parmi 28 sont bien classées par le système (FIS) initial.
- les observations considérées au départ de la classe 4 (diamètre du défaut (0.028inch)) qui correspondent aux lignes 85 à 112 dans la matrice des données, 3 observations de cette classe sur 28 sont bien classées par le système (FIS) initial.

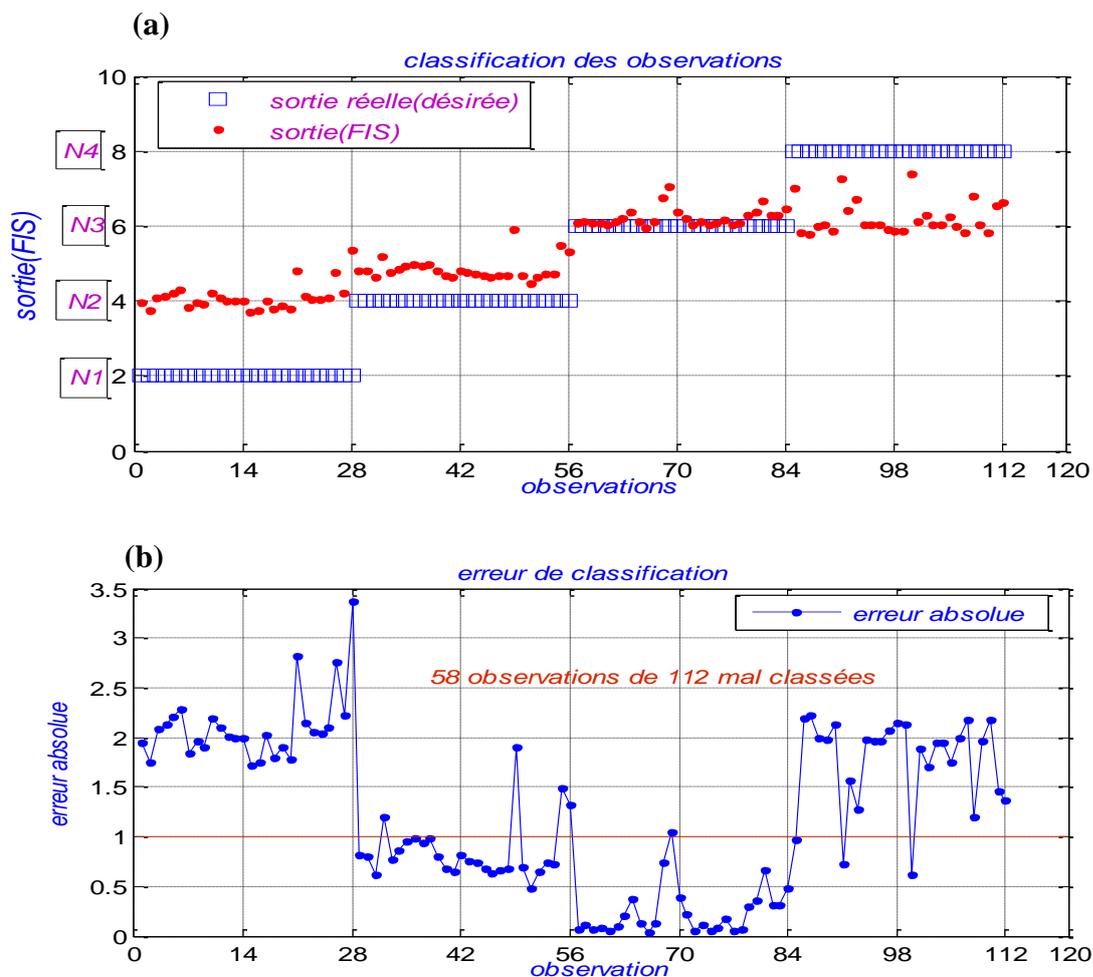


Figure 4.21 classification de la base de données. (a) sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée).

a.2) Interprétation du résultat de classification par le système (FIS) initial (avant l'optimisation)

Le système initial FIS pour cette étape (avant optimisation) n'a pas réussi à classer 58 observations sur 112, soit 48,21%. Ce rapport de bonne classification est faible, et ceci est principalement dû à la sélection aléatoire des entrées du système. Afin d'augmenter l'efficacité et la capacité de ce système dans le diagnostic et la classification des défauts, il convient d'optimiser ce dernier et c'est ce que nous allons faire dans les deuxième et troisième étapes.

4.2.3.2 Classification des défauts de roulements avec un système (FIS) optimisé structurellement (avec des variables d'entrées sélectionnées)

À partir des résultats de classification de la première étape obtenus par le système initial FIS, il a été constaté que ce système et avec la structure choisie est incapable de classer correctement la base de données. Donc il est nécessaire de modifier la structure du système et ses paramètres, et c'est ce que l'on appelle **optimisation**. Il s'agit de deux types d'optimisation, l'une structurelle et l'autre paramétrique.

Pour cette deuxième étape, on va optimiser notre système FIS **structurellement** en sélectionnant les paramètres d'entrée de ce dernier à l'aide d'une méthode de sélection automatique, contrairement à la première étape où on a choisie les paramètres d'une manière arbitraire.

Il existe plusieurs méthodes de sélection des paramètres. Dans notre travail, nous avons choisi la méthode de sélection par le **critère de Fisher**.

4.2.3.2.1 Sélection des indicateurs par le critère de Fisher

L'analyse de Fisher est une technique d'analyse factorielle discriminante, qui permet la projection de la base de données sur une surface informative de dimension réduite.

Contrairement à l'ACP (analyse en composante principale), il ne s'agit plus d'extraire les paramètres les plus représentatifs de la base de données, mais les paramètres les plus discriminants, vis-à-vis des classes représentées dans la base de données. Le problème est donc la détermination d'un sous ensemble d'axes discriminants ([Kudo et al, 2000](#)), ([Dubuisson, 1990](#)).

L'objectif est d'augmenter la séparabilité des classes. La détermination des axes discriminants est effectuée de manière à ce que la variation interclasses soit maximisée (dispersion des

classes entre elles) au regard de la variation intra-classes (dispersion à l'intérieur d'une classe) (casimir, 2003) et (Casimir et al, 2006).

D'une manière générale, pour M classes et pour un paramètre α , le critère de Fisher $J(\alpha)$ s'écrit :

$$J(\alpha) = \sum_{i=1}^M \sum_{j=1}^{M-1} \frac{(m_i(\alpha) - m_j(\alpha))^2}{N_i \sigma_i^2(\alpha) + N_j \sigma_j^2(\alpha)} \dots\dots\dots(4.1)$$

Avec: α : un paramètre parmi l'ensemble globale des paramètres.

m_i : Centre de gravité de la classe w_i .

σ_i^2 : Variance de la classe w_i .

N_i : Nombre de vecteurs dans la classe w_i .

Dans notre travail, on a 4 classes et 11 indicateurs, on peut calculer le critère de Fisher $J(\alpha)$ pour chaque indicateur (voir **Algorithme 4.1**), pour cela on a élaboré un programme sous environnement Matlab (**voir annexe 4.7**).

L'histogramme suivant montre la variation du critère de Fisher pour les 11 indicateurs (Figure 4.22).

- **Algorithme 4.1** *Calcul du critère de Fisher pour les différents indicateurs*

1 – Entrer la base de données **base_do**, $\text{size}(\text{base_do}) = [\text{nb_individus}, \text{nb_indicateurs}] = [112, 11]$

2 – Entrer le nombre de classe K , $\text{nb_class} = K$, $K \geq 2$

3 – Calculer la **moyenne(m)** et la **variance(V)** pour chaque classe K

$n=1$, $y = []$, $m = []$, $v = []$.

for $K=1:K$,

$Nw(K) = \text{Entree}(\text{'nombre d'individus de cette classe, il doit être } \geq 2\text{'})$;

$y = \text{base_do}(n:n+Nw(K)-1,:)$;

$m(K,:) = \text{mean}(y)$;

$V(K,:) = \text{VAR}(y, 1)$;

$n = n + Nw(K)$;

end

4 – Calculer la valeur du critère de Fisher **F(j)** pour les différents indicateurs j .

for $j=1: \text{nb_indicateurs}$,

$F(j) = 0$;

for $K=1: \text{nb_class}-1$,

for $l=K+1: \text{nb_class}$,

$\text{Num} = (m(K,j) - m(l,j))^2$;

$\text{Den} = Nw(K) * V(K,j) + Nw(l) * V(l,j)$;

$F(j) = F(j) + \text{Num}/\text{Den}$;

end

end

end

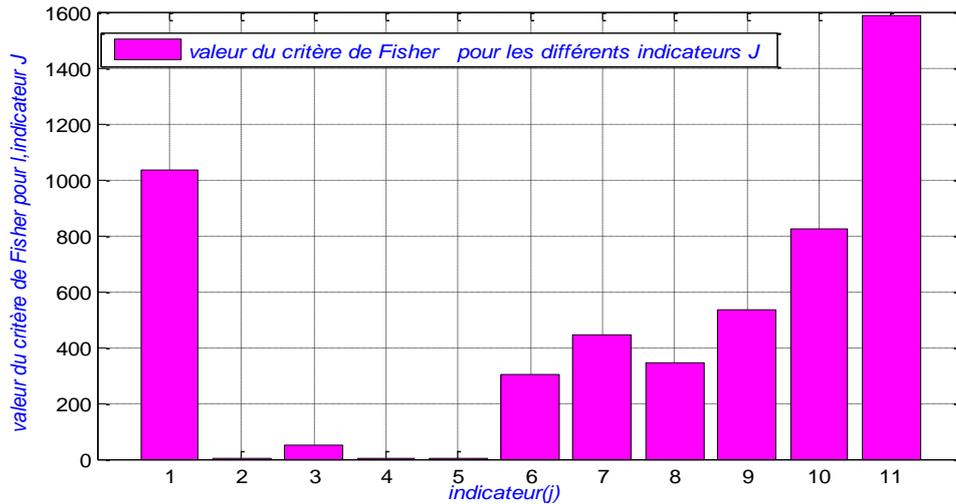


Figure 4.22 variation du critère de Fisher pour les 11 paramètres

La Figure 4.23 montre la variation du critère de Fisher pour les 11 paramètres selon un ordre décroissant.

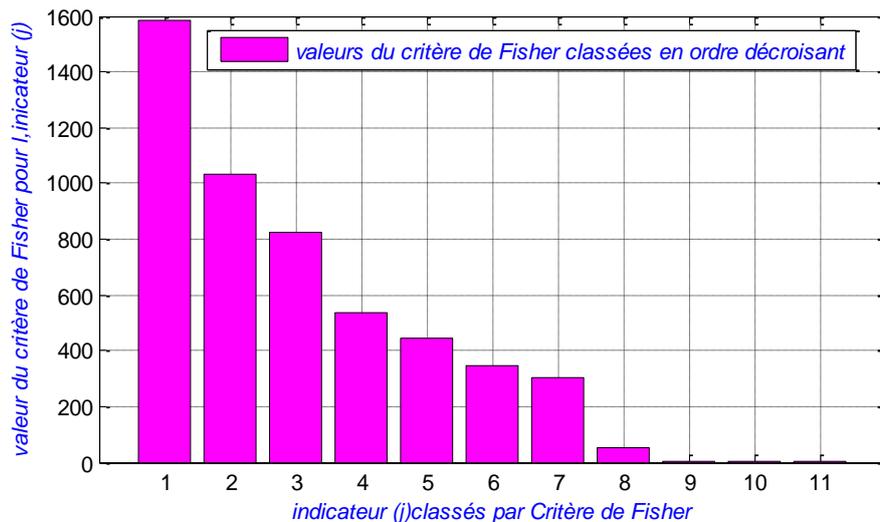


Figure 4.23 Critère de Fisher pour les différents indicateurs en ordre décroissant

Les indicateurs(j) de 1 à 11 dans la Figure 4.23 sont respectivement : E_{B5} , E_T , E_{B4} , E_{B3} , E_{B1} , E_{B2} , E_{BT} , A_{cc} , K_u , K_s , F_c .

Bien que le critère établi permette de classer les indicateurs calculés par ordre décroissant de puissance de discrimination, le nombre qui doit en être conservé ne peut être déterminé qu'à la suite d'essais pratiques par une règle de décision. Toutefois, pour remédier à ceci, on va utiliser le test d'arrêt qui consiste à déterminer la dimension correspondant à un pourcentage de perte d'informations admissible fixé a priori par l'utilisateur (l'expert) (Oukhellou, 1997).

L'expression (4.2) permet de calculer le pourcentage expliqué pour chaque paramètre du vecteur forme VF.

$$J_{N_j}(\%) = \frac{100 \times J(\alpha_j)}{\sum_{i=1}^d J(\alpha_i)} \quad j=1, \dots, d \quad \dots\dots\dots(4.2)$$

Les résultats obtenus sont représentés sur la Figure 4.24 :

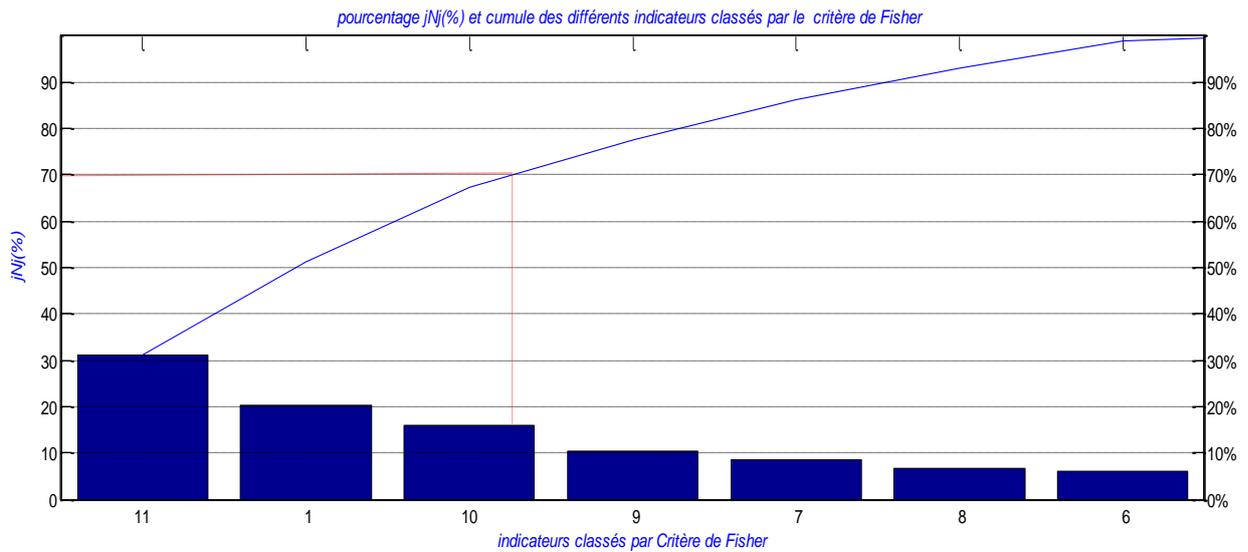


Figure 4.24 Pourcentage expliqué et cumule des différents indicateurs classés par le critère de Fisher

Si on considère qu'un seuil supérieur à 70(%) est largement acceptable, le vecteur forme sera donc de dimension $d' = 3$:

$$x = [EB5, ET, EB4]$$

Avec:

- EB5: Energie du spectre d'enveloppe dans la bande [4000-5000HZ]
- ET: l'écart-type(ET)
- EB4: Energie du spectre d'enveloppe dans la bande [3000-4000 kHz]

Remarque :

Les indicateurs dont les valeurs sont les plus élevées pour le critère de Fisher sont les plus pertinents.

De l'histogramme de la Figure 4.24, les trois premiers indicateurs qui ont des valeurs du critère de Fisher les plus élevées sont :

- EB5 : Energie du spectre d'enveloppe dans la bande [4000-5000HZ]

- ET : l'écart-type(ET)

- EB4 : Energie du spectre d'enveloppe dans la bande [3000-4000 kHz]

Ces trois indicateurs seront utilisés comme des variables d'entrée pour le système FIS utilisé dans cette deuxième étape.

4.2.3.2.2 Constitution du vecteur forme selon les indicateurs sélectionnés (vecteur d'indicateurs)

La composante du vecteur forme est basé sur le résultat de la sélection par le critère de Fisher (section 4.2.3.2.1). Ainsi, les trois indicateurs ci-dessus cités ont été considérés dans le vecteur forme et aussi du fait qu'ils ont permis une meilleure séparation des différentes classe de la base de données. Ces indicateurs représenteront les variables d'entrée du système FIS.

Les données que l'on doit classer et traiter sont stockées dans un tableau de type observations/variables de la forme suivante :

$$base - donn\acute{e}es = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1M} \\ x_{21} & x_{22} & & x_{2M} \\ \vdots & & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{NM} \end{pmatrix} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_N \end{bmatrix}$$

Où $\begin{cases} N \text{ est le nombre d'observations.} \\ M \text{ est le nombre de paramètres} \end{cases}$

On a alors :

M : indicateurs, représentés en colonnes, et N observations représentés en ligne qui seront divisés en N_c classes (w_1, w_2, \dots, w_{N_c}) ou modes de fonctionnement, Chaque classe w_i possède N_i observations.

Pour notre travail :

- N=112 observations (28 tranches par 4 signaux).

- M=3 ; nous avons choisie ces trois indicateurs pour cette deuxième étape d'une manière sélective (critère de Fisher) et ce contrairement à la première étape où le choix était arbitraire. Le choix sélectif des indicateurs a pour but l'optimisation structurelle du système (FIS) et espérer obtenir une meilleure classification que celle de la première étape. Les trois indicateurs choisis sont ; un de type statistique : l'écart type (ET), et les deux autres de type fréquentiel (E_{B5}) et (E_{B4}) qui représentent respectivement les énergies issue des bandes [4000-5000HZ], [3000-4000HZ] du spectre d'enveloppe.

Remarque : l'annexe (4.8) présente un tableau qui contient la base de données selon les conditions citées dans cette section.

4.2.3.2.3 Choix des classes

Dans cette deuxième étape (optimisation structurelle), nous utiliserons la même répartition que celle utilisée dans la première étape, voir section (4.2.2).

4.2.3.2.4 Implémentation du système d'inférence floue (FIS) optimisé structurellement sous Matlab

L'implémentation du système d'inférence floue (FIS) sous Matlab, peut être réalisée par deux méthodes différentes, soit à l'aide des commandes de la boîte à outils «Fuzzy Logic» ou par l'interface graphique de cette dernière.

a) Implémentation du système (FIS) optimisé structurellement par l'interface Graphique

a.1) Structure du système d'inférence floue (FIS) optimisé structurellement

Nous avons choisi, un système d'inférence floue FIS de type sugeno (voir Figure 4.25) avec trois variables à l'entrée (ET, EB₄, EB₅), et une seule variable à la sortie qui représente l'état de santé du système étudié (**état du système**).

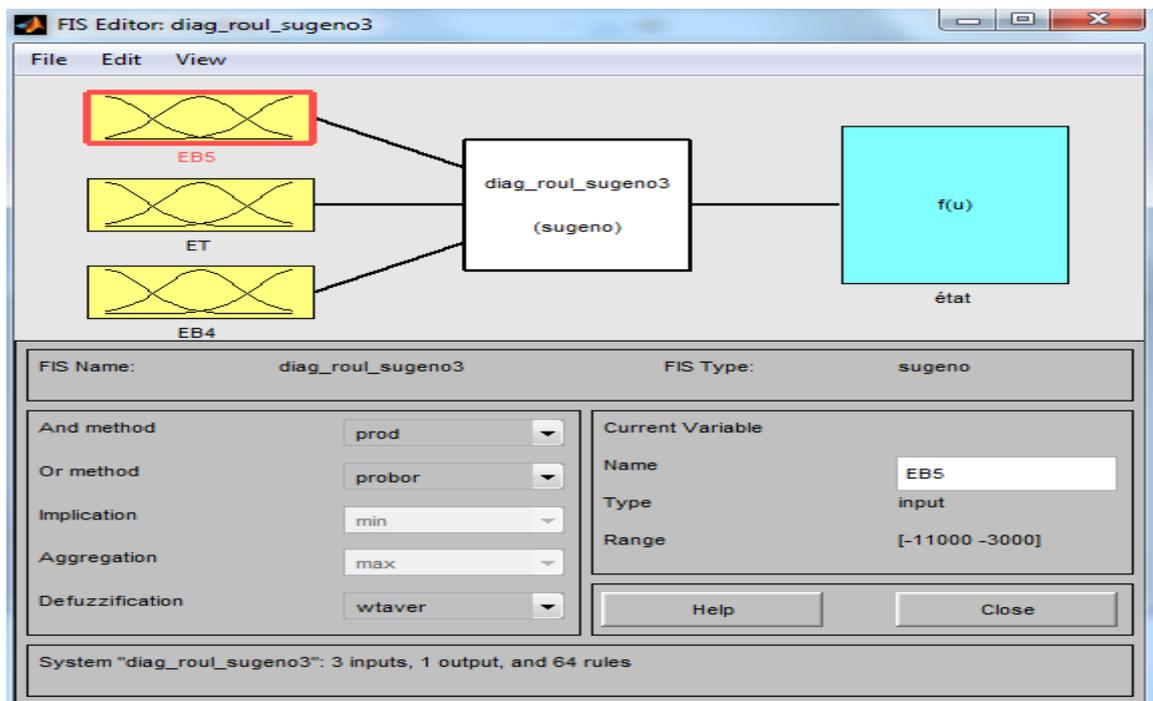


Figure 4.25 Schéma bloc du (SIF).

a.2) Fuzzification des variables d'entrée et de sortie

a.2.1) Fuzzification de la variable d'entrée «énergie de la bande [4000-5000HZ] (E_{B5}) »

Comme Cette variable varie dans l'intervalle [$\min(E_{B5}) = -1.0841 \times 10^4$, $\max(E_{B5}) = -3.5456 \times 10^3$] (voir [annexe \(4.8\)](#)). Donc, le domaine de variation [-1.1000×10^4 , -3.0000×10^3] a été choisi avec quatre fonctions d'appartenance de type gaussienne (une fonction gaussienne se caractérise par deux paramètres [σ , μ] où σ est l'écart type et μ est l'espérance mathématique). La première fonction appelée «très petite(TP) » avec deux paramètres [$\sigma = 28.3147$, $\mu = -1.0776 \times 10^4$], la deuxième fonction appelée «petite(P) » avec deux paramètres [$\sigma = 76.8611$, $\mu = -5.9030 \times 10^3$], la troisième fonction appelée «grande(G) » avec deux paramètres [$\sigma = 43.0092$, $\mu = -5.3566 \times 10^3$], et la dernière fonction appelée «très grande(TG)» avec deux paramètres [$\sigma = 33.5590$, $\mu = -3.6330 \times 10^3$]. La Figure 4.26 montre la fuzzification de la variable d'entrée «énergie de la bande [4000-5000] (E_{B5})».

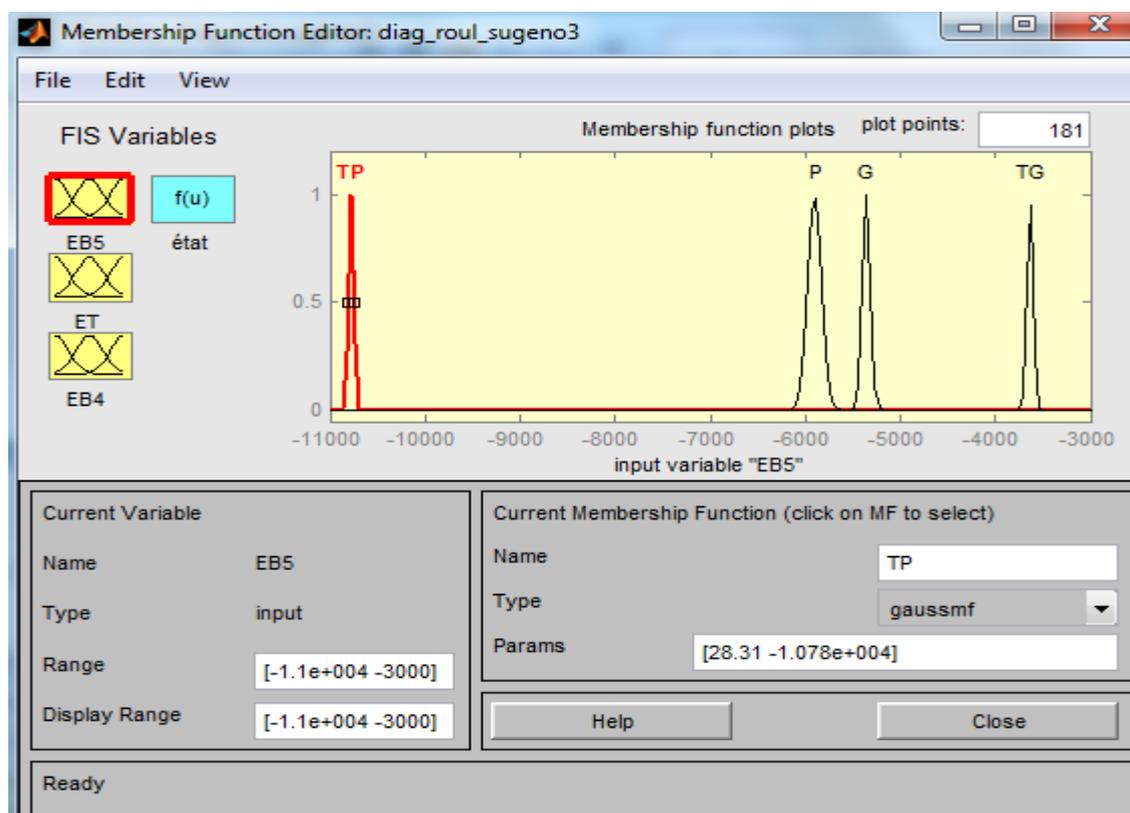


Figure 4.26 Fuzzification de la variable d'entrée «énergie de la bande [4000-5000] (E_{B5}) »

a.2.2) Fuzzification de la variable d'entrée «écart_type (ET)»

Cette variable varie dans l'intervalle [$\min(ET) = 0.0301$, $\max(ET) = 0.4193$] (voir [annexe \(4.8\)](#)). Donc on a choisi un domaine de variation de [0.0000 0.5000], et quatre fonctions d'appartenance de type gaussienne. La première fonction appelée «très petite(TP) » avec

deux paramètres [$\sigma= 3.3187 \times 10^{-4}$, $\mu= 0.0310$], la deuxième fonction appelée «petite(P)» avec deux paramètres [$\sigma= 0.0025$, $\mu= 0.1655$], la troisième fonction appelée «grande(G)» avec deux paramètres [$\sigma= 0.0041$, $\mu= 0.2167$], et la dernière fonction appelée «très grande(TG)» avec deux paramètres [$\sigma= 0.029$, $\mu= 0.4139$]. La Figure 4.27 montre la fuzzification de la variable d'entrée «écart_type (ET)»

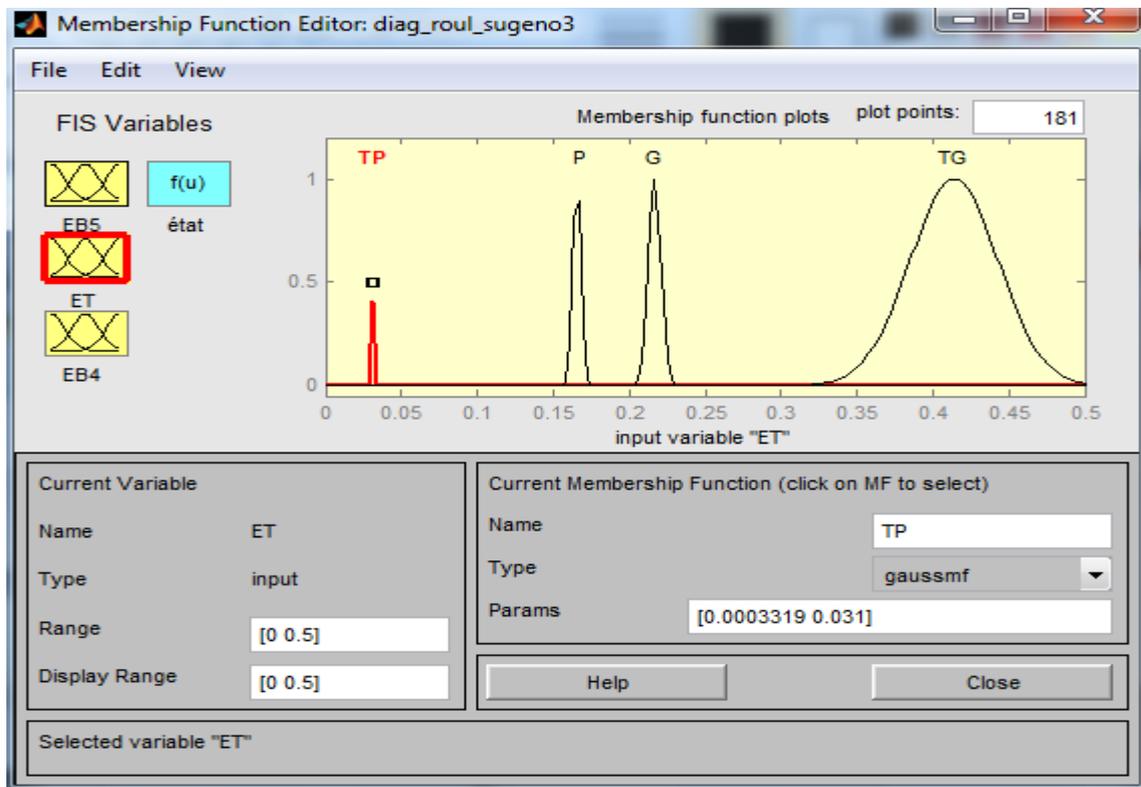


Figure 4.27 Fuzzification de la variable d'entrée «écart_type (ET)»

a.2.3) Fuzzification de la variable d'entrée «énergie de la bande [3000-4000HZ] (E_{B4})»

Cette variable varie dans l'intervalle [$\min(E_{B4})=-9.3824 \times 10^3$, $\max(E_{B4})=-3.8562 \times 10^3$] (voir annexe (4.8)). Donc on a choisi un domaine de variation de [-10000.0 -3000.0], et quatre fonctions d'appartenance de type gaussienne. La première fonction appelée «très petite(TP)» avec deux paramètres [$\sigma= 26.8731$, $\mu= -9.3280 \times 10^3$], la deuxième fonction appelée «petite(P)» avec deux paramètres [$\sigma= 53.6323$, $\mu=-6.3729 \times 10^3$], la troisième fonction appelée «grande(G)» avec deux paramètres [$\sigma= 51.7327$, $\mu= -6.0775 \times 10^3$], et la dernière fonction appelée «très grande(TG)» avec deux paramètres [$\sigma= 36.2670$, $\mu= -3.9624 \times 10^3$]. La Figure 4.28 montre la fuzzification de la variable d'entrée «énergie de la bande [3000-4000] (E_{B4})»

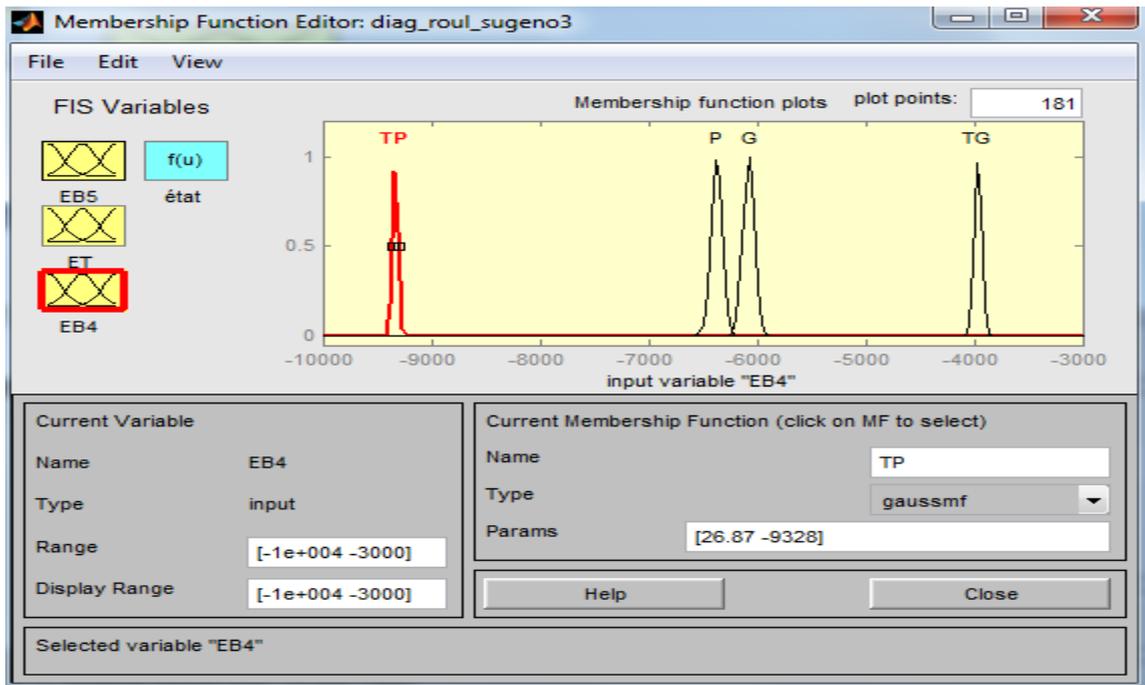


Figure 4.28 Fuzzification de la variable d'entrée « énergie de la bande [3000-4000HZ] (EB4) ».

a.2.4) Fuzzification de la variable de sortie « état du dispositif (état) »

On a choisi pour cette variable de sortie un domaine de variation [0, 10], et comme il ya 4 ensembles flous pour chacune des trois variables d'entrée, la variable de sortie sera définie sur 64 ensembles flous. Les fonctions d'appartenances représentant les différents ensembles flous sont toutes de type (constante). La Figure 4.29 montre la fuzzification de la variable de sortie « état du dispositif (état) ».

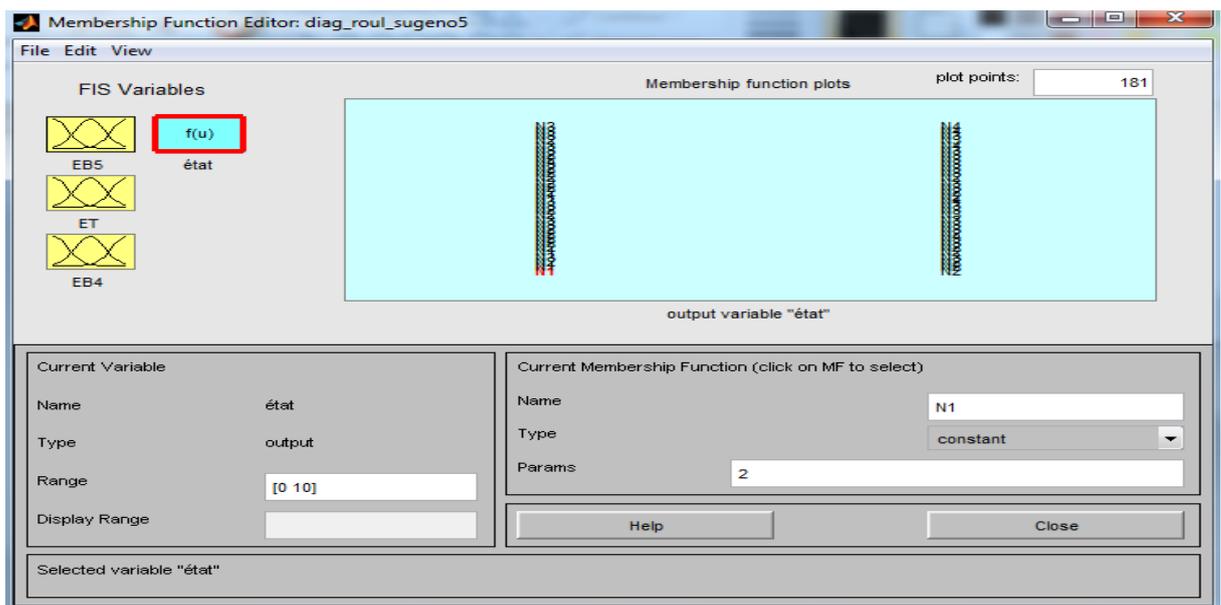


Figure 4.29 Fuzzification de la variable de sortie « état du dispositif (état) ».

Remarque :

Dans l'étape de fuzzification des variables d'entrée et de sortie. Nous avons choisi pour les variables d'entrée des fonctions d'appartenance de type gaussienne, Comme nous pouvons également choisir d'autres types (**trap ézo îles, trigonom étriques, etc.**). En ce qui concerne la variable de sortie, nous avons sélectionné des fonctions d'appartenance de type **constante**, comme nous pouvons aussi utiliser le type **lin éaire** (pour la fuzzification de la variable de sortie, le système FIS de type (Sugeno) fonctionne seulement avec deux types de fonctions d'appartenance (constante et linéaire))

a.3) Génération des règles d'inférence

Les différentes règles possibles selon la fuzzification des différentes variables d'entrée et de sortie choisis sont les suivantes (3 variables d'entrée et 4 fonctions d'appartenances pour chacune des variables, on obtient 64 règles d'inférences ($4^3=64$)):

1. si EB5 est TP et ET est TP et EB4 est TP alors état est N1
2. si EB5 est TP et ET est TP et EB4 est P alors état est N1
3. si EB5 est TP et ET est TP et EB4 est G alors état est N2
4. si EB5 est TP et ET est TP et EB4 est TG alors état est N3
5. si EB5 est TP et ET est P et EB4 est TP alors état est N1
6. si EB5 est TP et ET est P et EB4 est P alors état est N2
7. si EB5 est TP et ET est P et EB4 est G alors état est N2
8. si EB5 est TP et ET est P et EB4 est TG alors état est N2
9. si EB5 est TP et ET est G et EB4 est TP alors état est N2
10. si EB5 est TP et ET est G et EB4 est P alors état est N2
11. si EB5 est TP et ET est G et EB4 est G alors état est N3
12. si EB5 est TP et ET est G et EB4 est TG alors état est N3
13. si EB5 est TP et ET est TG et EB4 est TP alors état est N2
14. si EB5 est TP et ET est TG et EB4 est P alors état est N2
15. si EB5 est TP et ET est TG et EB4 est G alors état est N3
16. si EB5 est TP et ET est TG et EB4 est TG alors état est N3
17. si EB5 est P et ET est TP et EB4 est TP alors état est N1
18. si EB5 est P et ET est TP et EB4 est P alors état est N2
19. si EB5 est P et ET est TP et EB4 est G alors état est N2
20. si EB5 est P et ET est TP et EB4 est TG alors état est N2
21. si EB5 est P et ET est P et EB4 est TP alors état est N2

22. si EB5 est P et ET est P et EB4 est P alors état est N2
23. si EB5 est P et ET est P et EB4 est G alors état est N2
24. si EB5 est P et ET est P et EB4 est TG alors état est N3
25. si EB5 est P et ET est G et EB4 est TP alors état est N2
26. si EB5 est P et ET est G et EB4 est P alors état est N2
27. si EB5 est P et ET est G et EB4 est G alors état est N3
28. si EB5 est P et ET est G et EB4 est TG alors état est N3
29. si EB5 est P et ET est TG et EB4 est TP alors état est N2
30. si EB5 est P et ET est TG et EB4 est P alors état est N3
31. si EB5 est P et ET est TG et EB4 est G alors état est N3
32. si EB5 est P et ET est TG et EB4 est TG alors état est N3
33. si EB5 est G et ET est TP et EB4 est TP alors état est N2
34. si EB5 est G et ET est TP et EB4 est P alors état est N2
35. si EB5 est G et ET est TP et EB4 est G alors état est N2
36. si EB5 est G et ET est TP et EB4 est TG alors état est N3
37. si EB5 est G et ET est P et EB4 est TP alors état est N2
38. si EB5 est G et ET est P et EB4 est P alors état est N2
39. si EB5 est G et ET est P et EB4 est G alors état est N3
40. si EB5 est G et ET est P et EB4 est TG alors état est N3
41. si EB5 est G et ET est G et EB4 est TP alors état est N2
42. si EB5 est G et ET est G et EB4 est P alors état est N3
43. si EB5 est G et ET est G et EB4 est G alors état est N3
44. si EB5 est G et ET est G et EB4 est TG alors état est N3
45. si EB5 est G et ET est TG et EB4 est TP alors état est N3
46. si EB5 est G et ET est TG et EB4 est P alors état est N3
47. si EB5 est G et ET est TG et EB4 est G alors état est N3
48. si EB5 est G et ET est TG et EB4 est TG alors état est N4
49. si EB5 est TG et ET est TP et EB4 est TP alors état est N2
50. si EB5 est TG et ET est TP et EB4 est P alors état est N2
51. si EB5 est TG et ET est TP et EB4 est G alors état est N3
52. si EB5 est TG et ET est TP et EB4 est TG alors état est N3
53. si EB5 est TG et ET est P et EB4 est TP alors état est N2
54. si EB5 est TG et ET est P et EB4 est P alors état est N3
55. si EB5 est TG et ET est P et EB4 est G alors état est N3

56. si EB5 est TG et ET est P et EB4 est TG alors état est N3
 57. si EB5 est TG et ET est G et EB4 est TP alors état est N3
 58. si EB5 est TG et ET est G et EB4 est P alors état est N3
 59. si EB5 est TG et ET est G et EB4 est G alors état est N3
 60. si EB5 est TG et ET est G et EB4 est TG alors état est N4
 61. si EB5 est TG et ET est TG et EB4 est TP alors état est N3
 62. si EB5 est TG et ET est TG et EB4 est P alors état est N3
 63. si EB5 est TG et ET est TG et EB4 est G alors état est N4
 64. si EB5 est TG et ET est TG et EB4 est TG alors état est N4

a.4) Défuzzification de la variable de sortie

✓ Exemple d'application

On prend un exemple d'application pour une observation de la première classe (sans défaut(N1)) et on le classe ensuite par le système FIS de cette deuxième étape :

$X = [-1.0767 \times 10^4, 0.0310, -9.3080 \times 10^3]$ (voir la première ligne de la matrice de l'annexe (4.8)).

La sortie réelle du système FIS pour le vecteur X est : état=2

On remarque que notre système FIS a bien classé cette observation (état=2), elle est classée dans la première classe 1(N1).

Ces valeurs sont portées à titre d'exemple, le système proposé FIS accepte toutes les observations de la base de données. La Figure 4.30 montre la défuzzification pour l'exemple pratique donné

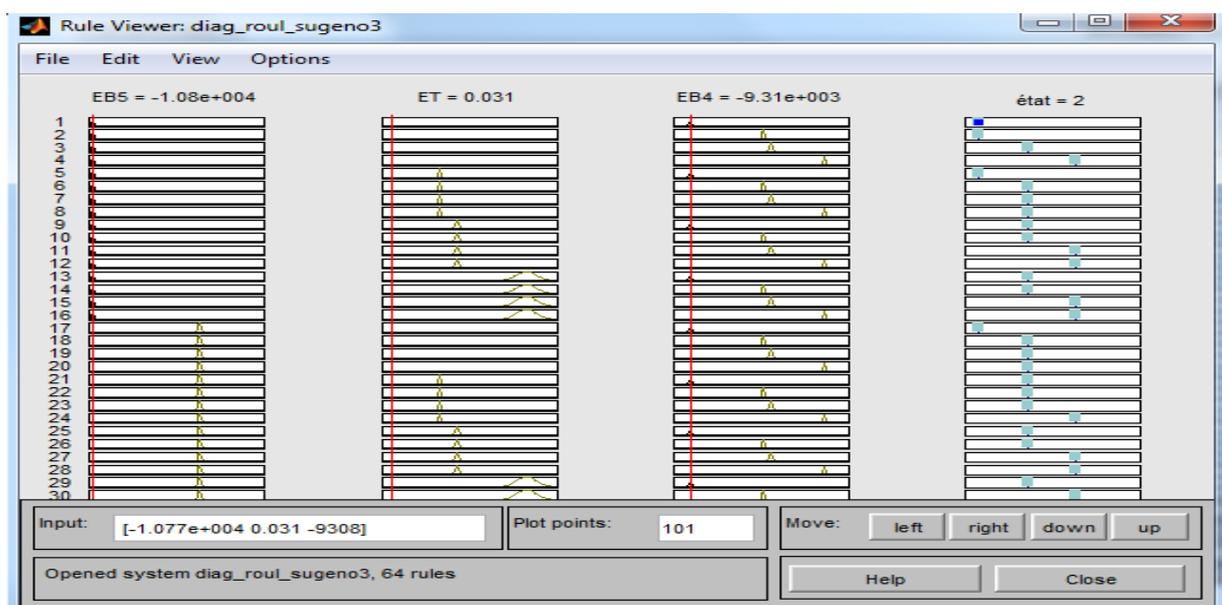


Figure 4.30 Défuzzification de la variable de sortie (état).

b) Implémentation du système(FIS) optimisé structurellement par les commandes de la boîte à outils «fuzzy logic »

Nous avons conçu un programme sous l'environnement Matlab à l'aide des commandes de la boîte à outils «fuzzy logic » et ce la pour l'implémentation du système d'inférence floue (FIS) optimisé structurellement et aussi pour la classification des défauts de roulements (le déroulement de l'implémentation est déjà présenté dans la section (3.4) du chapitre 3, Le programme développé est présenté dans **l'annexe (4.9)**.

4.2.3.2.5 Classification de la base de données par le système FIS (optimisé structurellement)

Le programme développé dans **l'annexe (4.9)**, permet de classer les observations de la base de données. **L'annexe (4.10)** présente le résultat de classification de la base de données.

Dans **l'annexe (4.10)**, on a :

En ligne: une seule ligne ; qui représente la variable de sortie.

En colonne : 112 colonnes ; portent les valeurs réelles de la variable de sortie (état du dispositif étudié (état)).

a) Discussion du résultat de classification

a.1) Représentation graphique du résultat de classification

La Figure 4.31.a représente graphiquement le résultat de classification, et la Figure 4.31.b montre l'erreur absolue de la sortie du système FIS optimisé structurellement par rapport à la sortie souhaitée (désirée), une observation est considérée comme mal classée lorsque l'erreur absolue pour cette dernière est supérieure ou égale à 1. On remarque que :

- les observations considérées au départ de la classe1 (sans défauts) qui correspondent de la ligne 1 à 28 dans la matrice des données sont bien classées par le système (FIS).
- les observations considérées au départ de la classe2 (diamètre du défaut (0.007inch)) qui correspondent de la ligne 29 à 56 dans la matrice des données sont bien classées par le système(FIS).
- les observations considérées au départ de la classe3 (diamètre du défaut (0.021inch)) qui correspondent de la ligne 57 à 84 dans la matrice des données sont bien classées par le système (FIS).
- les observations considérées au départ de la classe4 (diamètre du défaut (0.028inchh)) qui correspondent de la ligne 85 à 112 dans la matrice des données sont aussi bien classées par le système (FIS).

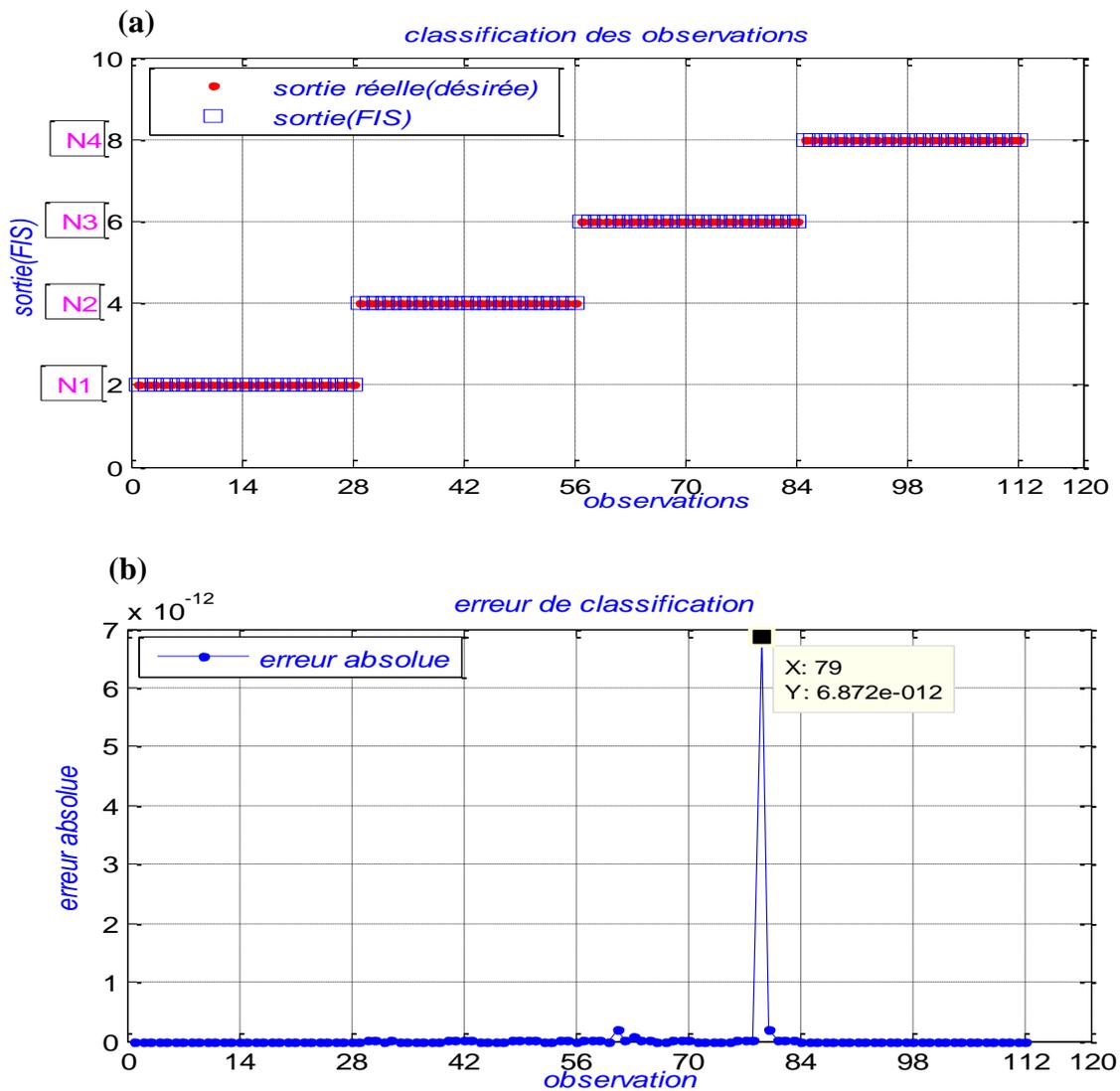


Figure 4.31 classification de la base de données. (a) sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée).

a.2) Interprétation du résultat de classification par le système FIS (optimisé structurellement)

Comme conclusion, On peut dire que notre système (FIS), optimisé structurellement a permis de classer correctement la totalité des observations de la base de données (c'est-à-dire un rapport de bonne classification de 100%) et cela est dû au bon choix des paramètres (indicateurs) d'entrée du système FIS qui sont les plus pertinents parmi l'ensemble des indicateurs. Pour comparer cette méthode avec d'autres méthodes d'optimisation, on procède à un deuxième type d'optimisation dans l'étape suivante (troisième étape), c'est l'optimisation paramétrique.

4.2.3.3 Classification des défauts de roulements avec un système FIS optimisé paramétriquement (à l'aide d'un système ANFIS)

Après avoir effectué la classification de la base de données de roulements dans la première étape par un système(FIS) où nous avons choisi ces paramètres d'entrée d'une manière arbitraire, les résultats de classification étaient loin des résultats souhaités. Lors de la deuxième étape, nous avons optimisé la structure du système(FIS) en choisissant les meilleurs indicateurs qui ont les valeurs les plus élevées du critère de Fisher comme entrées pour le système(FIS), ce qui a eu comme effet positif un taux de bonne classification de 100%. Dans la troisième étape, et pour des fins de comparaison avec d'autres méthodes d'optimisation, on procède à une optimisation paramétrique. Dans cet objectif, nous avons opté pour l'approche neuro-floue (hybridation de la logique floue avec les réseaux de neurones) en utilisant le système (ANFIS), qui intègre en même temps les avantages de la logique floue et ceux des réseaux de neurones. Ce système par son algorithme vise à ajuster les paramètres du système initial (FIS) par l'option d'apprentissage du système ANFIS, le but de l'ajustage est de chercher les meilleures valeurs pour les paramètres du système (FIS), qui donnent le meilleur résultat de classification. Dans cette dernière étape, nous abordons l'implémentation du système (ANFIS) sous environnement Matlab et la classification de la base de données.

4.2.3.3.1 Implémentation du système ANFIS (Adaptive Neuro-Fuzzy Inference Systems) sous Matlab

Nous présentons ici l'environnement Matlab qui a permis de mettre en œuvre le système ANFIS. Par ailleurs, les résultats que nous avons obtenus dans cette troisième étape seront comparés avec les résultats des deux étapes précédentes.

a) Structure du système ANFIS

La Figure 4.32 représente l'interface graphique du système ANFIS disponible dans la boîte à outils de la logique floue. Elle est assez simple à utiliser et peut être ouverte grâce à la commande *anfisedit* sur Matlab. Le contenu de cette interface, a déjà été présenté dans la section 3.5 du chapitre 3. Nous allons utiliser cette interface pour optimiser le système initial (FIS) du point de vue paramétrique, et ce par l'ajustage des différents paramètres préalablement choisis.

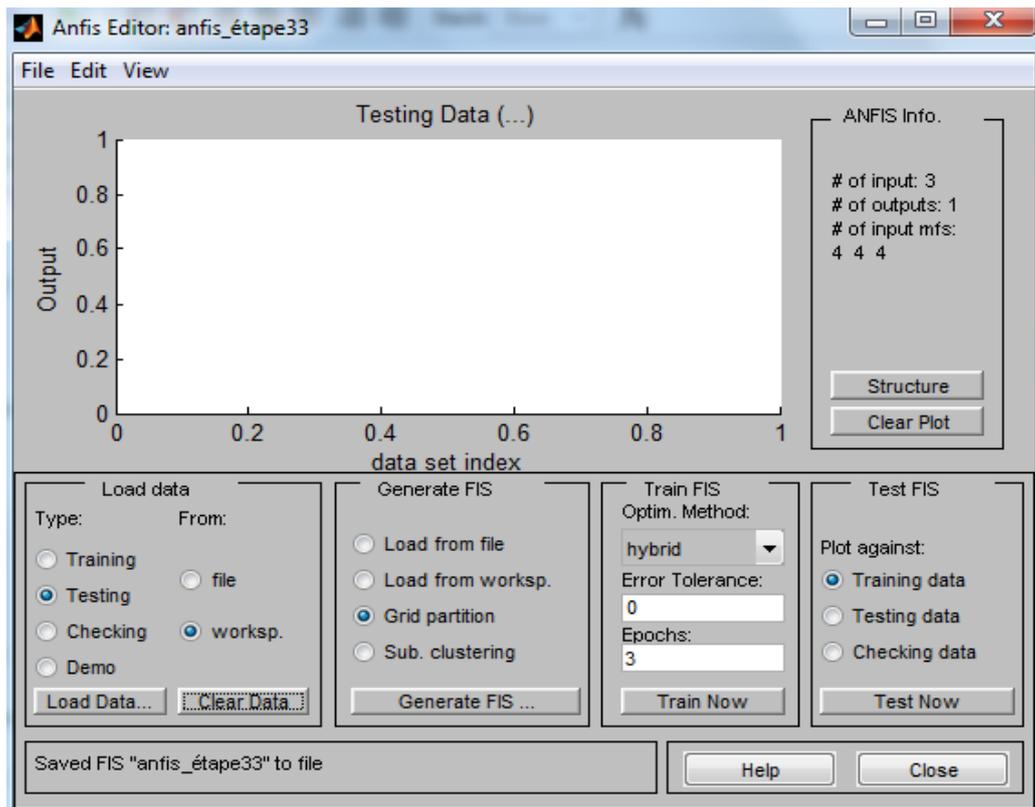


Figure 4.32 schéma bloc du système ANFIS

b) importation des bases de données (apprentissage et test)

On charge les données d'apprentissage et de test qui sont préparées à l'avance, les données sont des matrices dont les premières colonnes sont les entrées et la dernière colonne est réservée à la sortie. La taille de l'ensemble de données de base pour cette troisième étape est de (112×4) , elle est constituée par la base de données utilisée dans la deuxième étape qui a une taille de (112×3) à laquelle on ajoute une colonne à droite de taille (112×1) représentant les sorties désirées du système FIS. L'ensemble des données de base a été divisé en deux, une partie de 88 observations pour l'apprentissage et une autre de 24 observations réservée au test. Les Figures 4.33 et 4.34 représentent respectivement les données d'apprentissage et de test qui apparaissent dans la fenêtre centrale de l'interface ANFIS. Les données d'apprentissage sont représentées par des cercles (●) et les données de test par des astérisques (*).

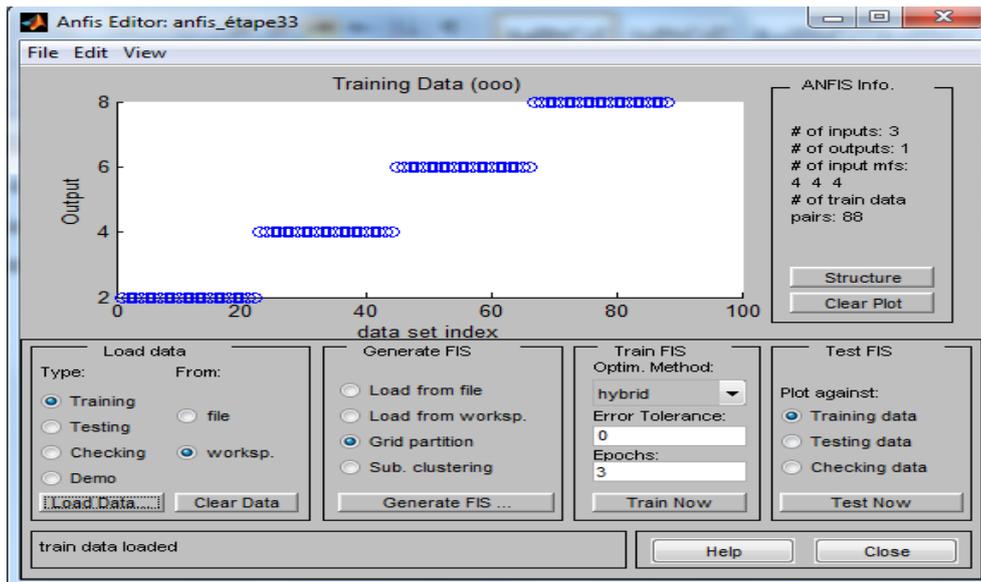


Figure 4.33 ensemble d'apprentissage

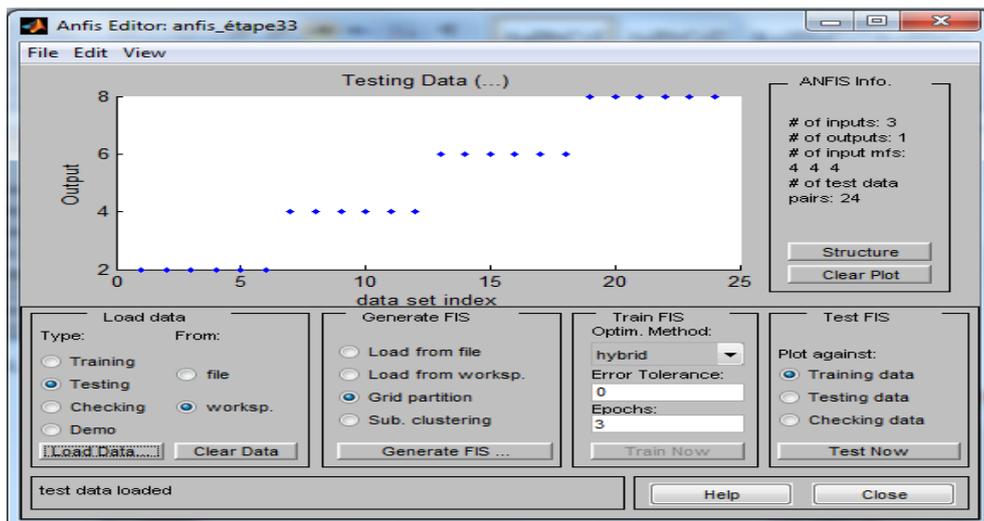


Figure 4.34 ensemble de test

c) Génération et initialisation du système initial (FIS)

Après avoir chargé les données d'apprentissage et de test, nous commençons à créer le système initial FIS que nous lui optimiserons ultérieurement. Le système ANFIS exécute cette tâche automatiquement, Il suffit de saisir le nombre de fonctions d'appartenance de chacune des trois entrées (les trois entrées sont les mêmes que les entrées du système FIS de la deuxième étape), le type de ces fonctions, ainsi que le type des fonctions d'appartenance de la sortie. La Figure 4.35 présente la boîte de dialogue dans laquelle ces paramètres sont fixés, Nous avons choisi 4 fonctions d'appartenance de type «gaussmf» pour chaque entrée et une fonction de type «constant» pour la variable de sortie. Après avoir validé les choix par le

bouton OK de cette boîte, le système ANFIS génère le système initial FIS, les Figures suivantes de 4.36 à 4.41 représentent les caractéristiques de ce dernier.

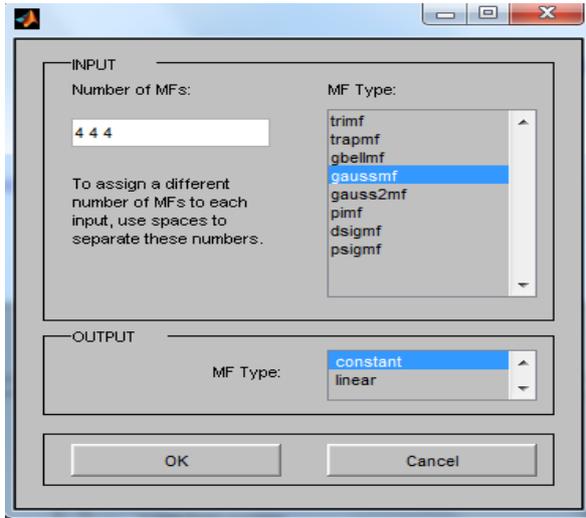


Fig 4.35 choix des paramètres du système initial SIF

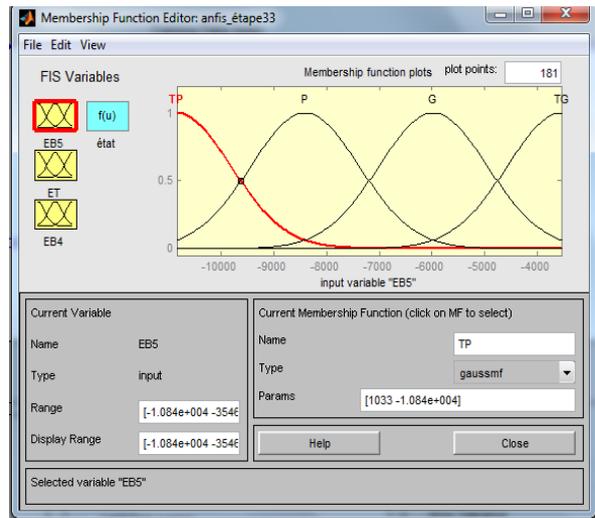


Fig 4.36 Fuzzification de la variable d'entrée «EB5»

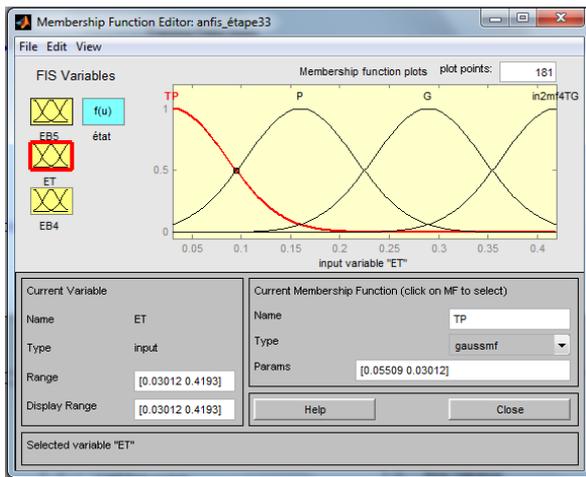


Fig 4.37 Fuzzification de la variable d'entrée «ET»

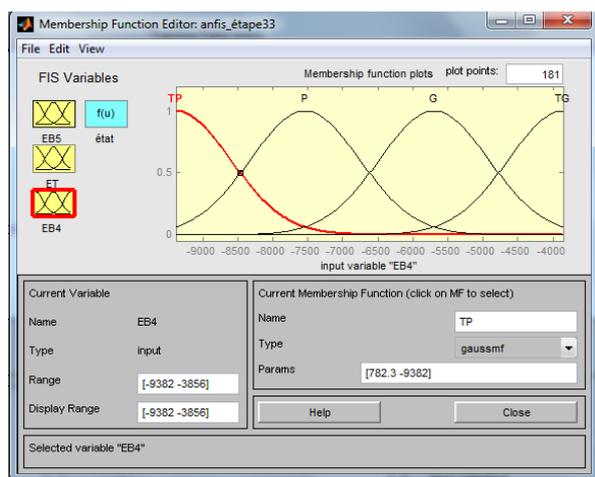


Fig 4.38 Fuzzification de la variable d'entrée «EB4»

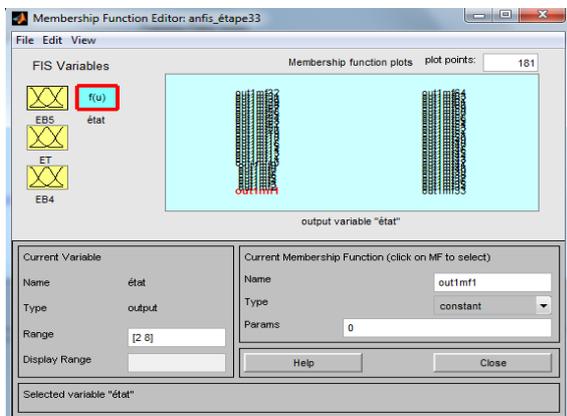


Fig 4.39 Fuzzification de la variable de sortie «état»

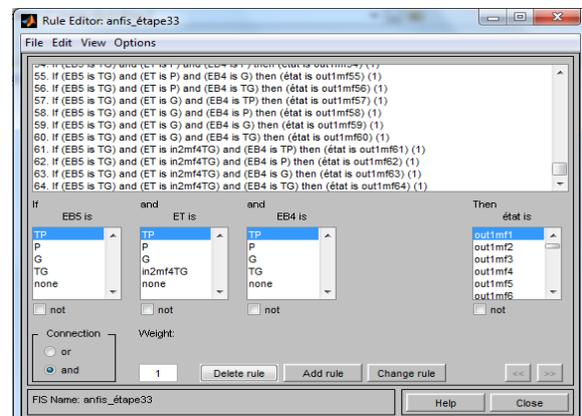


Fig 4.40 génération des règles floues

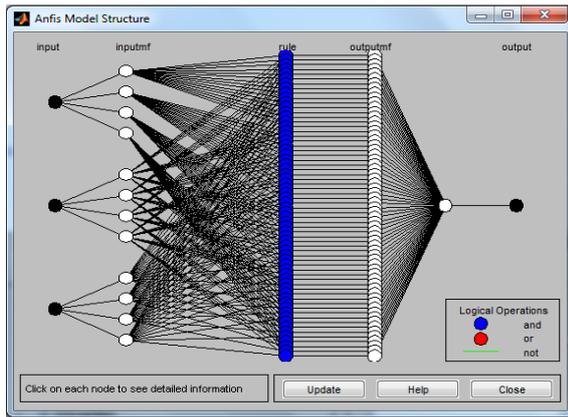
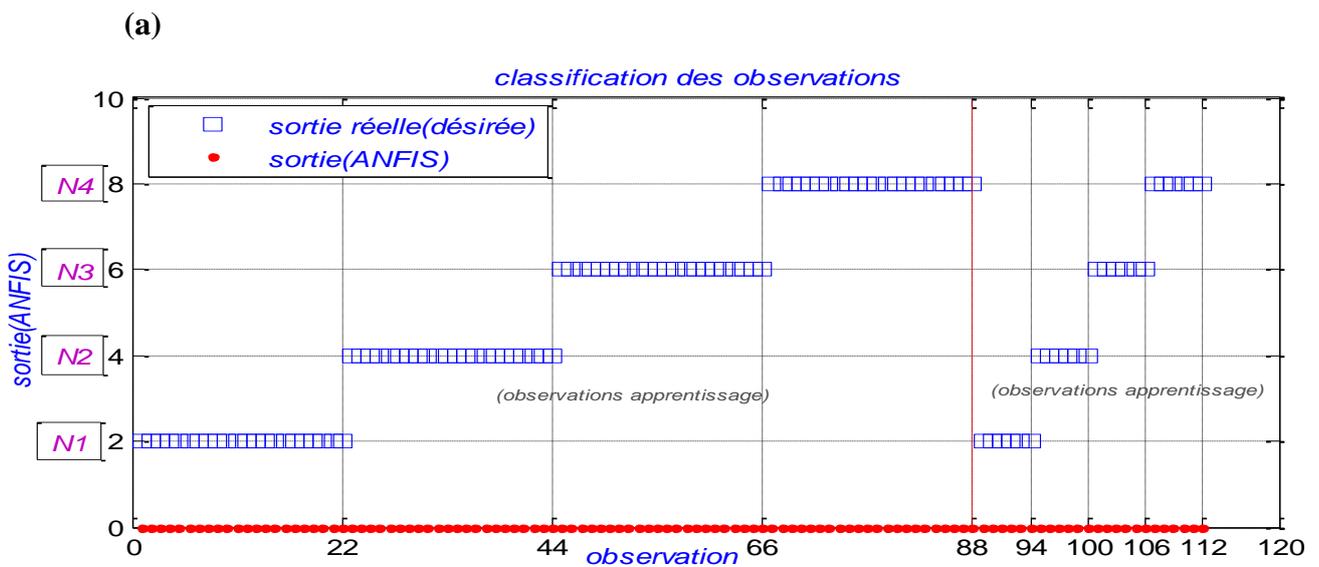


Fig 4.41 structure du système initial FIS

d) Classification des données d'apprentissage et de test à l'aide du système initial FIS (avant l'apprentissage)

Afin de comparer les résultats de classification avant et après le processus d'apprentissage, nous allons d'abord traiter la classification des données d'apprentissage et de test à l'aide du système initial FIS généré automatiquement par le système ANFIS (voir programme de classification dans l'annexe 4.11). Ensuite, et après le processus d'apprentissage, nous répétons le processus de classification à l'aide du nouveau système FIS (après apprentissage). Les Figures 4.42.a et 4.42.b montrent respectivement les résultats de classification des données d'apprentissage et de test et l'erreur absolue entre les résultats de classification par le système FIS (avant apprentissage) et les sorties du système souhaités (désirés).



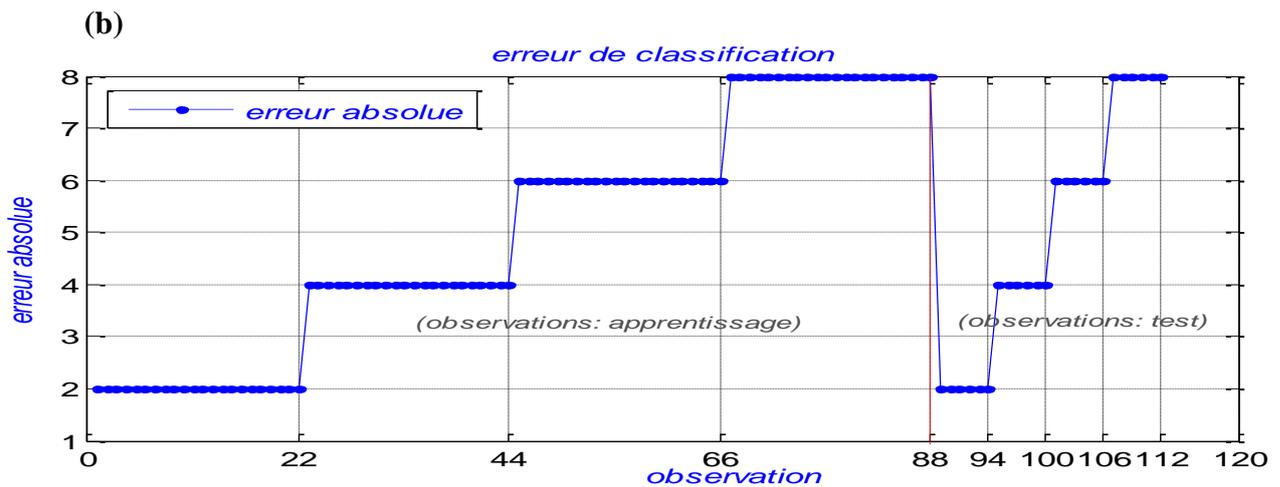


Figure 4.42 classification de la base de données (apprentissage et test). (a) sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée).

d.1) interprétation du résultat de classification du système initial FIS (avant optimisation)

Une observation est considérée mal classée si l'erreur entre la sortie du FIS, initial ici, et la sortie désirée est supérieure à 1. De ce fait et à partir des deux Figures 4.42.a et 4.42.b qui représentent respectivement les résultats de la classification et l'erreur absolue des données d'apprentissage et de test, on constate une mauvaise classification des observations. Ce résultat ne peut pas être jugé, car le système initial FIS n'a pas été conçu pour des tâches de classification, mais ce dernier est généré afin d'être optimiser par l'ajustage de ces différents paramètres à l'aide du système ANFIS, et ce afin d'obtenir un meilleur résultat de classification finale des données.

e) Apprentissage du système ANFIS

Au cours de cette étape, et après la génération du système initial FIS et le chargement des données d'apprentissage et de test, la phase d'apprentissage arrive et où il est question d'abord de sélectionner les critères nécessaires, comme il suit:

- Méthode d'apprentissage: méthode hybride (elle utilise les moindres carrés et la rétropropagation du gradient)
- tolérance d'erreur: 0 (utilisée pour créer un critère d'arrêt d'apprentissage, qui est lié à la taille d'erreur)
- Nombre d'itérations : 100(Nombre de cycles d'apprentissage)

Après avoir sélectionné ces critères, nous autorisons ANFIS à mener l'opération d'apprentissage, l'apprentissage va permettre d'ajuster les différents paramètres du système initial FIS, des données de références (apprentissage et test) nous permettrons de les choisir pour obtenir une sortie qui soit le plus en phase avec celle que nous recherchons.

L'apprentissage s'arrête lorsque que l'erreur de ce dernier satisfait la tolérance d'erreur choisie. Il est préférable de laisser l'erreur tendre vers zéro si nous ne savons pas comment l'erreur d'apprentissage peut se comporter, La Figure 4.43 montre l'erreur d'apprentissage en fonction du nombre d'itération. Le système ANFIS choisi les nouveaux paramètres des différentes fonctions d'appartenance du système FIS (entrées et sortie) à la fin du processus d'apprentissage. L'examen des fonctions d'appartenance initiale et finale indique que les fonctions d'appartenance peuvent être modifiées en fonction de l'état réel par le jeu de données d'apprentissage et le processus d'apprentissage. La Figure 4.45 montre les fonctions d'appartenance initiale avant l'apprentissage (Figure 4.45.a) et finale après l'apprentissage (Figure 4.45.b) de chaque entrée. Pour la seule sortie du système FIS, le système ANFIS a modifié toutes les valeurs des paramètres des fonctions d'appartenance constantes associées à la sortie qui étaient nulles (avant l'apprentissage) en des valeurs non nulles (après l'apprentissage), cette quantité de changements dans les fonctions d'appartenance finales indique le niveau de pertinence au niveau de la sortie, où un niveau de pertinence plus élevé implique plus de contribution à la classification(Voir Figure 4.44).

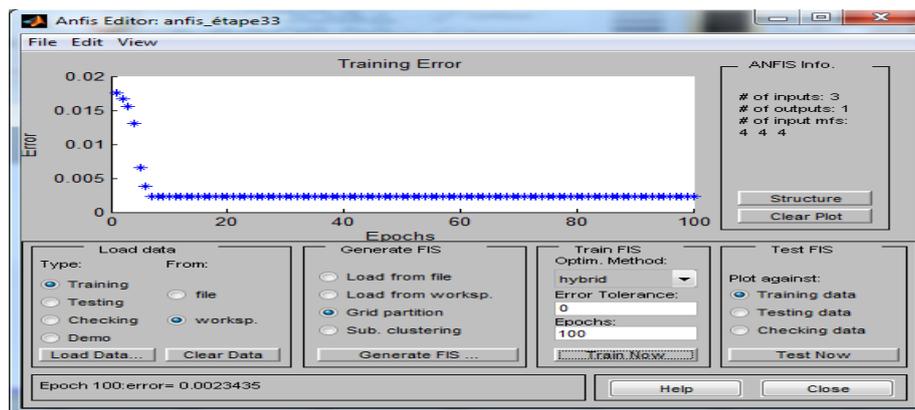


Figure 4.43 erreur d'apprentissage du système ANFIS

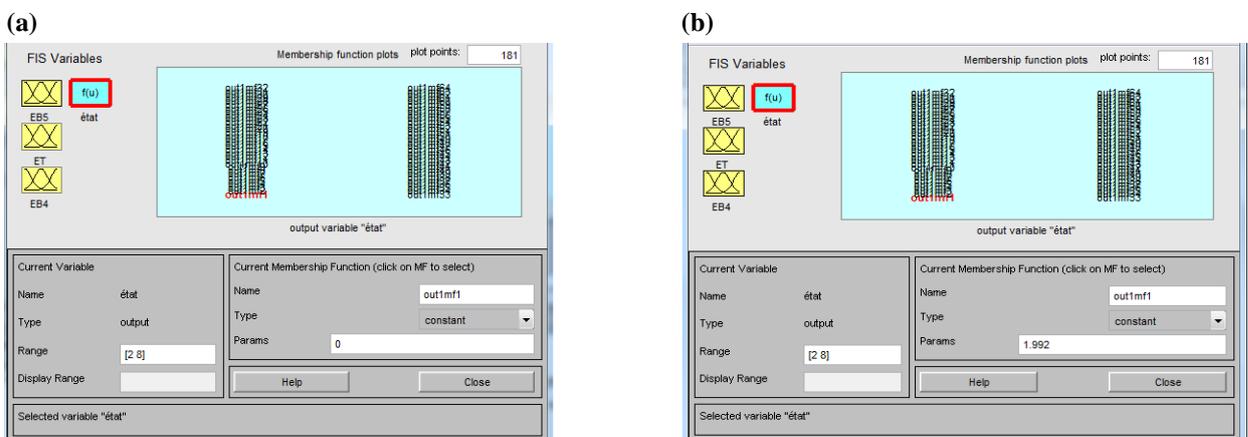


Figure 4.44 Fonctions d'appartenance de la seule sortie avant l'apprentissage (colonne de gauche (a)) et après l'apprentissage (colonne de droite(b)) où il y a un changement pertinent au niveau des valeurs de paramètres.

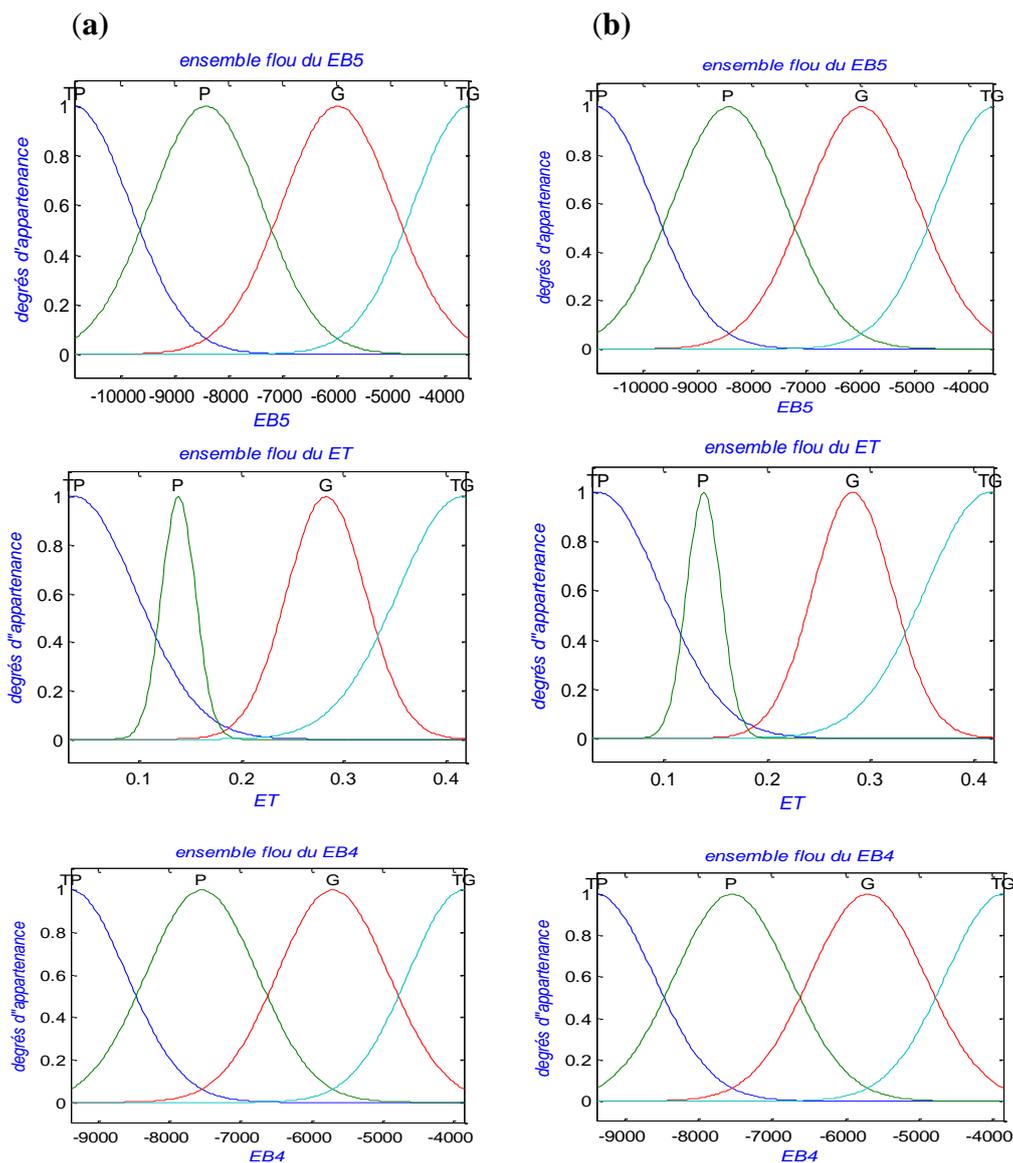


Figure 4.45 Fonctions d'appartenance de chaque entrée avant l'apprentissage (colonne de gauche (a)) et après apprentissage (colonne de droite (b)) où l'axe vertical de chaque sous-figure représente le degré d'appartenance.

f) Test du système ANFIS entraîné

Après que le système ANFIS ait été entraîné, les fonctions d'appartenance finales sont utilisées pour déterminer la valeur en degré de chaque entrée de la phase de test. L'ensemble de données de test sert à valider la précision du modèle ANFIS pour le diagnostic des défauts de roulements. Le but de la classification est d'affecter une observation d'entrée à l'une des 4 classes concernées par la présente étude et représentées par les étiquettes de classification définies (étiquette 2 : classe 1(Niveau1), étiquette 4 : classe 2(Niveau2), étiquette 6 : classe 3(Niveau3), étiquette 8 : classe 4(Niveau4)). Le résultat de classification obtenu par l'approche proposée ANFIS pour le jeu de données d'apprentissage et de test est présenté dans la Figure 4.46. La Figure 4.46.a montre la sortie désirée et la sortie de l'ANFIS pour

l'ensemble d'apprentissage et la Figure 4.46.b montre la sortie désirée et la sortie de l'ANFIS pour l'ensemble de test.

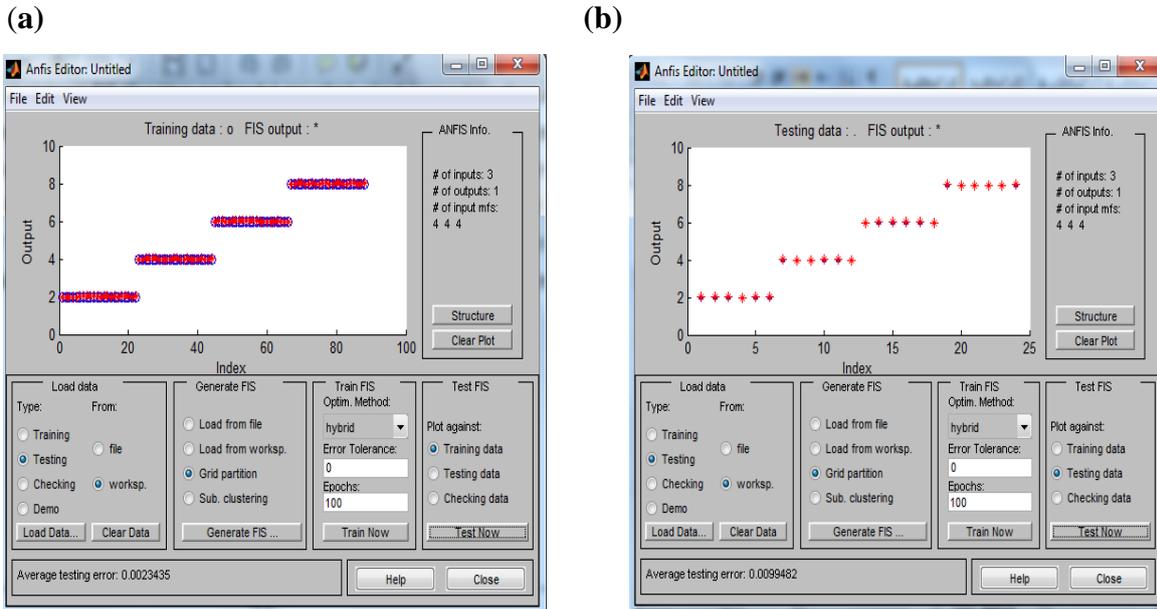


Figure 4.46 Résultats de la classification avec l'approche proposée ANFIS après l'apprentissage, (a) résultat d'apprentissage avec 88 échantillons; (b) résultat de test avec 24 échantillons.

4.2.3.3.2 Classification des données d'apprentissage et de test par le système (FIS) optimisé paramétriquement par le système (ANFIS)

Après l'apprentissage du système ANFIS, nous allons à nouveau reclassifier la base de données (apprentissage et test), mais cette fois-ci avec le nouveau système FIS optimisé paramétriquement par ANFIS (voir annexe 4.12), et ce pour savoir dans quelle mesure le système s'améliorera en termes de capacité à classifier correctement. Nous utiliserons les mêmes bases de données d'apprentissage et de test que celles utilisées dans la classification par le système initial (avant l'apprentissage). Ces deux bases de données comprennent 88 observations pour l'apprentissage et 24 observations pour le test. En plus de la classification des observations des deux bases de données, nous allons également calculer l'erreur absolue dans cette classification. L'erreur absolue dans la classification de toute observation est définie comme étant la différence en valeur absolue entre la valeur de la sortie du système FIS (après apprentissage) et la valeur de la sortie souhaitée. Pour décider si une observation est classée correctement ou non, on se base sur le fait que si l'erreur absolue de classification d'une observation est supérieure ou égale à un (1), alors l'observation est mal classée autrement, elle est bien classée. La Figure 4.47.a représente la classification des bases de

données d'apprentissage et de test par le système FIS optimisé (après apprentissage) et la Figure 4.47.b représente l'erreur absolue dans la classification de ces deux base de données.

Remarque : l'annexe 4.13 présente le résultat de classification de la base de données de roulements (apprentissage et test) par le système d'inférence floue FIS optimisé par le système ANFIS.

Pour Cette annexe, on a :

En ligne (une seule ligne) : sortie du système FIS (état)

En colonne (de 1 à 12) : ces colonnes portent des valeurs réelles de la sortie du système FIS (résultats de classification des observations de la base de données (apprentissage de 1 à 88 et le test de 89 test)).

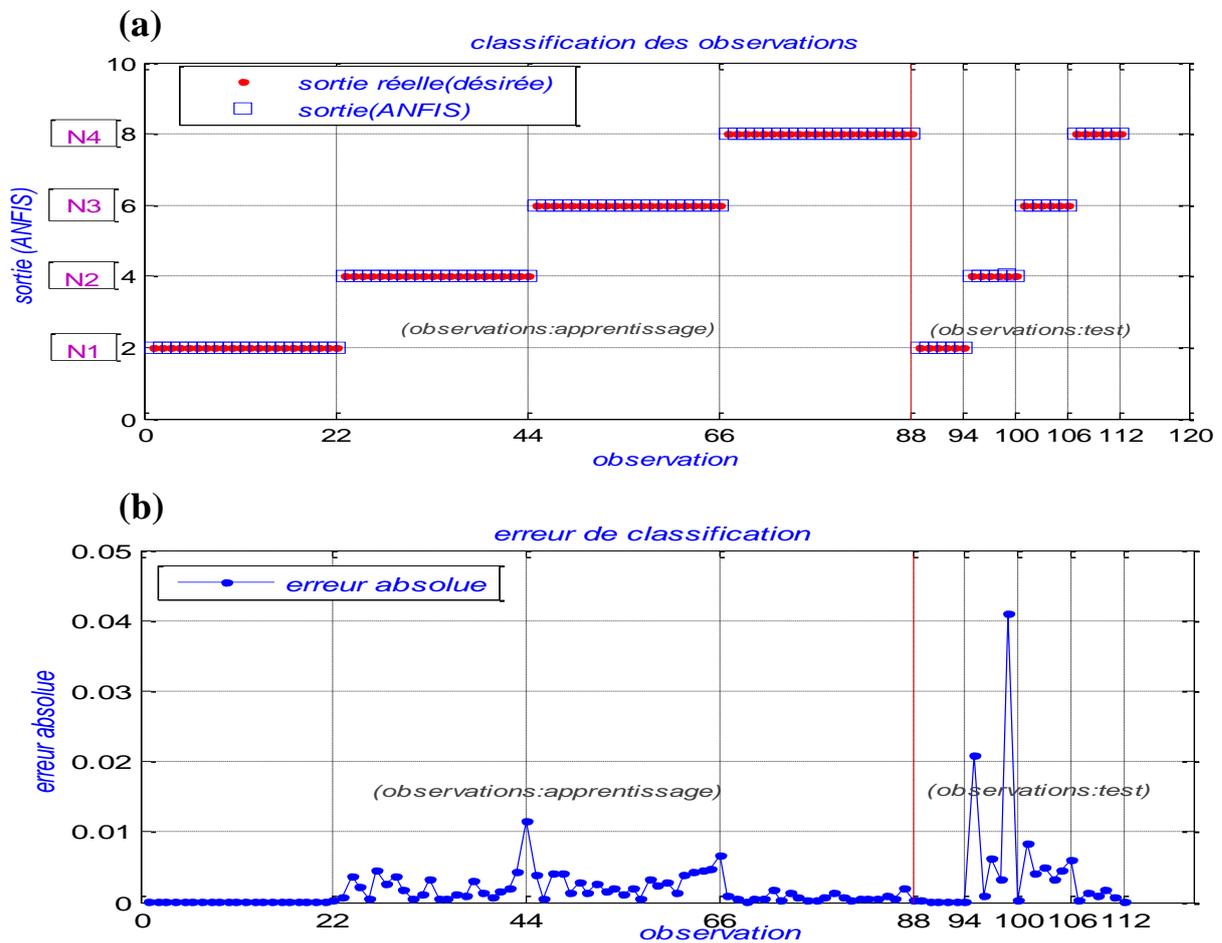


Figure 4.47 classification de la base de données (apprentissage et test) avec le système FIS optimisé par ANFIS .
 (a) sortie du système FIS ; (b) l'erreur absolue de la sortie du système FIS par rapport à la sortie souhaitée (désirée).

a) Interprétation du résultat de classification par le système (FIS) optimisé paramétriquement par le système (ANFIS)

À partir des résultats de classification représentés par les Figures 4 47.a et 4 47. b, l'erreur des observations des deux bases de données est inférieure à 1. De ce fait, on peut dire que avec le système FIS après apprentissage, toutes les observations sont bien classées. Le rapport de bonne classification est de 100%. Ce taux de 100% démontre la capacité du système ANFIS dans l'optimisation paramétrique du système FIS initial bien que ce taux était nul avant apprentissage. Ce taux élevé est aussi la conséquence de l'hybridation du système flou avec les réseaux de neurones formant un système d'apprentissage supervisé. Ce système hybride bénéficie de la capacité de la logique floue dans la prise de décision et celle des réseaux neuronaux dans l'apprentissage et l'adaptation. Les résultats expérimentaux montrent que l'approche proposée (ANFIS) ne peut pas uniquement discriminer de manière fiable les différentes catégories de défauts mais, également identifier le niveau de gravité du défaut. Ainsi, l'approche proposée ANFIS offre la possibilité de porter un diagnostic sur les défauts naissants.

4.3 Conclusion

La démarche développée pour optimiser le système de diagnostic des défauts de roulements (système à base d'inférence floue) comprend trois étapes :

Dans la première étape, le système initial d'inférence floue (FIS) a été créé. Ce dernier est du type Sugeno avec trois entrées et une seule sortie. Ces entrées, ont été sélectionnées de manière aléatoire parmi l'ensemble des indicateurs disponibles. La sortie du système représente l'état de santé du système étudié. Le choix du nombre et du type des fonctions d'appartenance associées aux différentes entrées et sorties, ainsi que la génération de la base des règles floues sont basées sur les données et les connaissances disponibles sur le système étudié. L'implémentation du système a été effectuée sous environnement MATLAB. Les résultats de classification et de diagnostic par ce système initial (FIS), ont conduit à un taux de bonne classification de l'ordre de 48,21%. Ce faible pourcentage est principalement dû à la sélection aléatoire des entrées du système, ce qui nécessite une optimisation de ce dernier afin d'améliorer son efficacité et sa capacité de classification. Pour cette raison, dans la deuxième étape, nous avons optimisé la structure du système en sélectionnant les entrées de ce dernier par une méthode de sélection automatique par le critère de Fischer qui permet de classer les différents indicateurs en fonction de leur pouvoir séparateur pour les différentes classes de la

base de données. Trois indicateurs ayant les plus grandes valeurs de critère du Fisher ont été choisis comme entrées du système FIS de cette deuxième étape. Le système d'inférence floue considéré est identique à celui de la première étape. Les résultats obtenus dans ce cas sont très encourageants du fait que le taux de bonne classification est 100%. Ce pourcentage élevé est dû à l'efficacité de la méthode de sélection automatique utilisée, qui nous a fourni les meilleurs indicateurs ayant conduit à une nette séparation des différentes classes de la base de données. Ce taux peut également s'expliquer par la bonne sélection des divers paramètres du système FIS (nombre et type de fonctions d'appartenance associées aux différentes variables d'entrées et de sorties du système, génération de la base de règles floues, ..., etc.).

Au cours de la troisième étape, et pour des fins de comparaison avec d'autres méthodes d'optimisation. On procède à une optimisation paramétrique, et pour cela un système hybride combinant d'un côté la logique floue, et d'un autre côté les réseaux de neurones est appliqué ce système hybride appelé (ANFIS), ce dernier ajuste les différents paramètres du système FIS que nous souhaitons optimiser pour obtenir un nouveau système capable de classifier et identifier les différents défauts de la base de données avec des résultats satisfaisants. Le système ANFIS s'appuie à la fois sur les avantages d'apprentissage et d'adaptation découlant des caractéristiques des réseaux de neurones, ainsi que de la capacité de traitement d'informations incertaines et vagues de la logique floue. L'implémentation du système ANFIS est effectué par l'interface graphique Anfis Editor de la boîte à outils «fuzzy logic», cela commence avec la génération du système initial (FIS) à optimiser, où nous avons autorisé le système ANFIS à le générer automatiquement à partir de la base de données d'apprentissage et de test composées successivement de 88 et de 24 observations, vient ensuite le processus d'apprentissage au cours duquel le système ANFIS ajuste et sélectionne les différents nouveaux paramètres pour le nouveau système optimisé d'inférence floue (FIS). En ce qui concerne les résultats de classification et du diagnostic à cette étape, et après que le taux de bonne classification était de zéro (0%) par le système FIS initial (avant l'apprentissage), ce dernier a atteint 100% après l'apprentissage, ce rapport élevé est le résultat de l'efficacité et de la capacité du système ANFIS dans l'ajustage et le choix des meilleures paramètres pour le nouveau système d'inférence floue (FIS), capable de classer les divers défauts du système étudié.

En raison de ce qui précède, il est possible de dire que nous avons optimisé le système d'inférence floue initial créé dans le but de diagnostiquer et de classer les défauts de roulements de deux manières différentes: La première touche la structure du système (structurelle) et la deuxième touche les valeurs des différents paramètres du système

(paramétrique). Et comme comparaison entre les deux méthodes. Les résultats de classification étaient bons et encourageants pour les deux méthodes. En ce qui concerne l'implémentation du système est plus facile avec la méthode utilisée dans la troisième étape et qui dépend du système ANFIS, parce que cette dernière est une méthode automatique, où nous ne sélectionnons que le nombre et le type de fonctions d'appartenance pour les entrées et sorties, tandis que la méthode utilisée dans la deuxième étape dépend fortement de l'expert à toutes les étapes du développement du système de diagnostic. Le temps requis pour l'implémentation est moins avec la méthode de la troisième étape (ANFIS), le temps d'implémentation n'est pas important pour le cas de notre travail (le temps est important pour le cas de diagnostic des systèmes en temps réel).

***CONCLUSION GENERALE ET
PERSPECTIVES***

Conclusion générale et perspectives

Les travaux menés dans le cadre de cette thèse portent sur la maintenance de systèmes industriels complexes utilisant la technique de la logique floue, qui est l'une des techniques de l'intelligence artificielle. L'objectif principal de ce travail est de trouver des solutions aux divers inconvénients des systèmes de diagnostic automatisés qui reposent sur la logique floue dans leur construction, ou, en d'autres termes, d'optimiser ces systèmes pour en accroître l'efficacité et les performances. Le test et la validation des systèmes de diagnostic construits ont été réalisés sur des défauts de roulements d'un banc d'essais.

Afin d'aborder le sujet, nous avons consacré le premier chapitre aux systèmes d'inférence floue, où nous avons présenté les divers concepts sur lesquels repose la théorie des ensembles flous, ainsi que les divers outils mathématiques nécessaires pour les manipuler. Nous avons également présenté les diverses étapes de construction des systèmes d'inférence floue (FIS), ainsi que leurs caractéristiques et les différentes méthodes pour les optimiser.

Dans le deuxième chapitre, nous avons présenté les principaux travaux et applications de certaines techniques d'intelligence artificielle dans le domaine du diagnostic des défauts mécaniques des machines tournantes, ainsi que les principaux avantages et inconvénients de ces techniques. Les travaux présentés concernent les techniques suivantes: la logique floue(LF), les réseaux de neurones artificiels (RNA), les systèmes neuro-flou(NF), les systèmes d'inférence neuro-flou adaptatif(ANFIS). Cet état d'art nous a permis d'un côté de savoir à quel niveau d'application ces techniques sont arrivées et quels sont les problèmes les plus importants qu'elles rencontrent, et d'un autre côté, nous avons pu définir le cadre dans lequel nous avons travaillé, et qui est principalement axé sur l'optimisation des systèmes de diagnostic basés sur les modèles d'inférence floue.

Le troisième chapitre présente la méthodologie suivie dans le travail expérimental lors de la réalisation de ce travail. Nous avons présenté les différentes étapes de la mise en œuvre des systèmes d'inférence floue (FIS) ainsi que les systèmes d'inférence neuro-flou adaptatif(ANFIS). Les programmes d'implémentation des différents systèmes établis au cours de ce travail étant sous l'environnement Matlab. Nous avons également présenté le

banc d'essais de roulements qui est la source des signaux vibratoires utilisés pour le test et la validation des divers programmes de diagnostic de défauts établis.

Dans le quatrième et dernier chapitre, nous avons présenté les différents résultats obtenus ainsi que leurs discussions et interprétation. L'objectif principal de ce travail étant d'optimiser un système de diagnostic des défauts mécaniques dans les machines tournantes, et qui dépend dans leur construction aux modèles d'inférence floue. Pour cela, nous avons donc choisi une méthodologie basée sur trois étapes.

Au cours de la première étape, nous avons construit un système initial d'inférence floue (FIS) en sélectionnant au hasard les entrées de ce dernier à partir d'un ensemble d'indicateurs préparés à l'avance suite à un prétraitement et une étude préliminaire des différents signaux disponibles enregistrés à partir d'un banc d'essais de roulements. Pour les résultats de classification de la base de données avec l'utilisation de ce système initial, le taux de bonne classification était de 48,21%. Ce faible taux s'explique principalement par la manière de sélection des entrées de ce système effectuée aléatoirement. Afin d'augmenter l'efficacité et la performance de ce système de diagnostic initial, il est nécessaire de l'optimiser, et c'est ce que nous avons fait aux deuxième et troisième étapes.

Au cours de la deuxième étape, nous avons optimisé le système de diagnostic du point de vue structurel (optimisation structurelle). L'optimisation apportée dans cette étape touche les entrées du système, Contrairement à la première étape, nous avons choisi les meilleurs indicateurs parmi la gamme d'indicateurs disponibles pour les utiliser comme entrées du nouveau système de diagnostic de cette étape, Ces indicateurs choisis et à travers le système de diagnostic doivent pouvoir séparer nettement les différentes classes représentant les différents défauts étudiés. Pour cela, nous avons utilisé une méthode de sélection appelée critère de Fisher, cette méthode permette de classer les différents indicateurs calculés par ordre décroissant de puissance de discrimination entre les différentes classes de défauts de chaque indicateur. Pour les résultats de classification avec le système de diagnostic optimisé structurellement de cette deuxième étape, le taux de bonne classification était de 100%, Ce taux élevé est principalement dû à la sélection que nous avons fait sur l'ensemble globale des indicateurs disponibles, les trois premiers indicateurs ont été choisies après le processus de sélection comme entrées pour le système de diagnostic, ce qui a donné au système une grande capacité et efficacité pour séparer nettement les différentes classes de défauts de la base de données.

Lors de la troisième et dernière étape, et après une optimisation structurelle du système, une autre méthode a été utilisée, qui s'applique à l'aspect paramétrique du système (optimisation paramétrique). Le but d'utilisation de cette méthode est de chercher le meilleur jeu de valeurs des différents paramètres du système d'inférence floue et qui donne le meilleur résultat de classification par rapport aux résultats souhaités pour la base de données représentant les différents défauts. À cette fin, nous avons utilisé un système intégrant la logique floue (LF) avec les réseaux de neurones artificiels (RNA), appelé ANFIS. Ce système, et grâce à la fonction d'apprentissage qui découle des avantages de réseaux de neurones artificiels et à l'aide d'une base de données dédiée, les valeurs des différents paramètres du système d'inférence floue initial sont ajustées. Le système d'inférence floue final obtenu une fois le processus d'apprentissage terminé est un système optimisé paramétriquement et qui sera en mesure de donner un meilleur résultat de classification de la base de données représentant les différents défauts étudiés. Le résultat de classification obtenu en utilisant le système de diagnostic optimisé de cette dernière étape était très élevé et avec un taux de 100%, ce résultat explique l'efficacité et la capacité de ce système hybride (ANFIS) à ajuster les différents paramètres du système d'inférence floue initial afin d'obtenir une meilleure configuration pour le système d'inférence floue finale et qui est capable de donner le résultat souhaité dans la classification des différents défauts étudiés. Ce taux élevé peut également s'expliquer par la nature du système intégré (ANFIS), qui combine à la fois les avantages de la logique floue et des réseaux de neurones, la logique floue avec sa capacité à prendre des décisions dans un environnement d'informations imprécises et inexacts et les réseaux de neurones avec ses capacités d'apprentissage et d'adaptation.

À titre de comparaison entre les deux méthodes utilisées pour optimiser le système de diagnostic des défauts, et en termes d'implémentation, la deuxième méthode (optimisation Paramétrique) prend moins de temps que la première (optimisation structurelle), Ceci est dû au fait que la méthode d'optimisation paramétrique utilisant le système hybride (ANFIS) est presque automatique à toutes les étapes dans leur implémentation à partir de l'initialisation du système d'inférence floue initial et en passant par le processus d'apprentissage avec une base de données dédiée (base de données d'apprentissage), jusqu'à l'obtention de la configuration finale du système d'inférence floue suivie par le test et la validation de ce dernier. Alors que la méthode d'optimisation structurelle prend plus de temps et nécessite une expertise plus spécialisée au cours de toutes les étapes d'implémentation et qui sont effectuées de manière classique (sélection des meilleurs indicateurs à l'aide d'une méthode de sélection choisie pour les utiliser comme entrées du système d'inférence floue, choix du type et nombre de fonctions

d'appartenance associés aux entrées et sorties, génération de la base de règles floues, le choix de la méthode de défuzzification, ...,etc).

En ce qui concerne les résultats de classification, pour chacune des deux méthodes d'optimisation, les résultats étaient très excellents et avec un taux de 100% et on peut indiquer ici une autre raison qui a contribué à la hausse de ce taux en plus de ce qui a été mentionné sur les avantages du système (ANFIS) inspiré à la fois des avantages de la logique floue et les réseaux de neurones, il s'agit de la géométrie des défauts étudiés qui sont des trous créés sur la bague intérieure des roulements et avec des dimensions de diamètres relativement espacées de trou à l'autre, Cela, et en particulier après l'optimisation des systèmes de diagnostic, a contribué au fait que la séparation entre les différentes classes représentant les différents défauts est relativement facile, les classes sont complètement séparées les unes des autres et ne se chevauchent pas.

Enfin, il convient de noter que les processus d'optimisation, et dans certains cas de diagnostic de défauts, doivent inclure à la fois les deux aspects structurels et paramétriques afin d'obtenir les résultats souhaités du processus de diagnostic.

Les travaux réalisés dans le cadre de cette thèse pourraient donner lieu à des études supplémentaires sur plusieurs points :

Le système construit dans ce travail dans le but de diagnostic des défauts mécaniques basé sur la logique floue peut être optimisé structurellement de plusieurs autres manières, tels que:

- utilisez d'autres méthodes de prétraitement des signaux pour développer de nouveaux indicateurs et élargir l'ensemble global des indicateurs, parmi ces méthodes (Empirical Mode Decomposition (EMD), Ensemble Empirical Mode Decomposition (EEMD), Multi-Scale Entropy (MSE), ... ,etc).
- utilisez d'autres méthodes pour classer et sélectionner l'ensemble d'indicateurs disponibles, et faire une comparaison entre ces méthodes sur la base des résultats de classification obtenus pour chaque méthode, parmi ces méthodes (le critère de Fisher régularisé (CFR), Improved Distance Evaluation (IDE), construction des arbres de décision par l'algorithme J48,etc).
- Modifiez le type et le nombre de fonctions d'appartenance utilisées pour les différentes variables d'entrées et de sorties et comparez entre eux à la base des résultats de classification de chaque type de fonction d'appartenance utilisé afin de rechercher le meilleur type de fonction d'appartenance.
- optimisez la base de règles floues en réduisant leur nombre et en éliminant les termes redondants afin de réduire le temps de calcul et l'exploitation du système flou de façon

optimale (technique du groupage flou (**Hohensohn, 94**), orthogonal least square **5(chen, 89)**, **(chen, 91)**).

- utilisez d'autres méthodes de défuzzification et comparez entre eux en cherchant la meilleure méthode qui donne les meilleurs résultats de classification.

Le système de diagnostic établi à base de la logique floue peut également être optimisé du côté paramétrique à l'aide des algorithmes génétiques.

REFERENCES
BIBLIOGRAPHIQUES

Références bibliographiques

(Abbasion et al, 2007) Abbasion, S., Rafsanjani, A., Farshidianfar, A., and Irani, N. 2007, "Rolling Element Bearings Multi-fault Classification Based on the Wavelet Denoising and Support Vector Machine", *Mechanical Systems and Signal Processing*, vol. 21, no. 7, pp. 2933-2945.

(Abu- Mahfouz, 2005) Abu-Mahfouz, I.A. 2005, "A Comparative Study of Three Artificial Neural Networks for the Detection and Classification of Gear Faults", *International Journal of General Systems*, vol. 34, no. 3, pp. 261-277.

(Aliustaoglu et Ertunc, 2009) Cuneyt Aliustaoglu, H. Metin Ertunc, Hasan Ocak, " Tool wear condition monitoring using a sensor fusion model based on fuzzy inference system ", *Mechanical Systems and Signal Processing* (2009).

(Altmann, 1999) Altmann, J. (1999) " Application of Discrete Wavelet Packet Analysis for the Detection and Diagnosis of Low Speed Rolling-Element Bearing Faults ". *PhD Thesis, Monash University: Melbourne*.

(Altmann et Mathew, 2001) Altmann, J., and Mathew, J. 2001, "Multiple Band-pass Autoregressive Demodulation for Rolling-element Bearing Fault Diagnosis", *Mechanical Systems and Signal Processing*, vol. 15, no. 5, pp. 963-977.

(Altug et al, 1999) Altug, S., Chow, M.-Y., and Trussell, H.J. 1999, "Fuzzy Inference Systems Implemented on Neural Architectures for Motor Fault Detection and Diagnosis", *IEEE Transactions on Industrial Electronics*, vol. 46, no. 6, pp. 1069-1079.

(Baillie et Mathew, 1994) Baillie, D., and Mathew, J. 1994, "Diagnosing Rolling Element Bearing Faults with Artificial Neural Networks", *Acoustics Australia*, vol. 22, no. 3, pp. 79-84.

(Baillie et Mathew, 1996) Baillie, D C and Mathew, J (1996), "A comparison of autoregressive modelling techniques for fault diagnosis of rolling element bearings" *Journal of Mechanical Systems and Signal Processing*, Vol 10(1), pp 1-17.

- (Bandemer, 1995)** Bandemer, S.Gottwald, "Fuzzy sets, fuzzy logic, fuzzy methods with applications", *Edition Wiley, England*, 1995.
- (Bay et Bayir, 2005)** Bay, O.F., and Bayir, R. 2005, "Kohonen Network Based Fault Diagnosis and Condition Monitoring of Pre-engaged Starter Motors", *International Journal of Automotive Technology*, vol. 6, no. 4, pp. 341-350.
- (Bezd et Sank, 1992)** James C. BEZDEK and Sanker K. PAL, "Fuzzy Models for Pattern Recognition", *IEEE Press*, 1992.
- (Boutros et Liang, 2007)** Tony Boutros, Ming Liang, "Mechanical fault detection using fuzzy index fusion ", *International Journal of Machine Tools and Manufacture* 47(2007) 1702-1714.
- (Breneur, 2002)** Breneur Claire, "Elements de maintenance préventive de machines tournantes dans le cas de défauts combinés d'engrenage et de roulements" , *Thèse de L'INSA de Lyon*, Décembre 2002, pp.120.
- (Brotherton, et al, 2000)** Brotherton, T., Jahns, G., Jacobs, J., and Wroblewski, D. 2000, "Prognosis of Faults in Gas Turbine Engines", *IEEE Aerospace Conference Proceedings 2000*, vol. 6, pp. 163-171
- (Cantor, 1884)** G. Cantor, " De la puissance des ensembles parfaits de points ", *Acta Math.*, vol. 4, 1884, p. 381-392.
- (Carpenter et al., 1992)** Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J.H., and Rosen, D.B. 1992, "Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps", *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 698-713.
- (Casimir, 2003)** R. Casimir, "Diagnostic des défauts des machines asynchrones par reconnaissance des formes " , *Thèse de L' école Doctorale de Lyon*, 2003, pp 172.
- (Casimir et al., 2006)** R. Casimir, E. Boutleux, G. Clerc et A. Yahoui "The use of features selection and nearest neighbors rule for faults diagnostic in induction motors" , *Engineering Applications of Artificial Intelligence*, 19, 2006, pp 169-177
- (Castejon et al, 2010)** Castejon, C., Lara, O., and Garcia-Prada, J.C. 2010, "Automated Diagnosis of Rolling Bearings Using MRA and Neural Networks", *Mechanical Systems and Signal Processing*, vol. 24, no. 1, pp. 289-299.

(Celik et Bayir, 2007) Celik, M.B., and Bayir, R. 2007, "Fault Detection in Internal Combustion Engines Using Fuzzy Logic", *Proceedings of the Institution of Mechanical Engineers*, vol. 221, no. D5, pp. 579-587.

(Chen et al, 2016) Xihui Chen, Gang Cheng, Hongyu Li, Min Zhang, "Diagnosing planetary gear faults using the fuzzy entropy of LMD and ANFIS", *Journal of Mechanical Science and Technology (JMST)*, June 2016, Volume 30, Issue 6, pp 2453–2462

(Chinnam et Baruah, 2004) Chinnam, R.B., and Baruah, P. 2004, "A Neuro-fuzzy Approach for Estimating Mean Residual Life in Condition-based Maintenance Systems", *International Journal of Materials and Product Technology*, vol. 20, no. 1-3, pp. 166-179.

(Dawu et al, 2009) Jian-Da Wu, Chuang-Chin Hsu, "Fault gear identification using vibration signal with discrete wavelet transform technique and fuzzy-logic inference", *Expert Systems with Applications* 36(2009) 3785-3794.

(Donat et al, 2008) Donat, W., Choi, K., An., W., Singh, S. and Pattipati, K. 2008, "Data Visualization, Data Reduction and Classifier Fusion for Intelligent Fault Diagnosis in Gas Turbine Engines", *Journal of Engineering for Gas Turbines and Power (Transactions of the ASME)*, vol. 130, no. 4.

(Dubuisson, 1990) B. Dubuisson, " *Diagnostic et reconnaissance des formes* ". *Trait é des nouvelles technologies. Série diagnostic et Maintenance. Editions Herm ès*, Paris 1990, pp 320.

(Earld, 1997) EARL D.COX, T.MAURICE CLERC, "la logique floue- pour les affaire et l'industrie". Thomson publishing. Paru le 23/1/1997.

(Felkaoui, 2007) Ahmed FELKAOUI. (2007). "Diagnostic vibratoire au service de la maintenance : de l'acquisition à la décision", *Thèse de doctorat d'état UFA S éif 1* 2007.

(Fedala, 2005) Semchedine Fedala, " le diagnostic vibratoire automatis é: comparaison des méthodes d'extraction et de sélection du vecteur forme", *M émoire de magister DOMP-UFAS, S éif* 2005, pp.142.

(Gao et Ovaska, 2001) Gao, X.Z., and Ovaska, S.J. 2001, "Soft Computing Methods in Motor Fault Diagnosis", *Applied Soft Computing*, vol. 1, no. 1, pp. 73-81.

(Garga et al, 2001) Garga, A.K., McClintic, K.T., Campbell, R.L., Yang, C.-C., Lebold, M.S., Hay, T.A., and Byington, C.S. 2001, "Hybrid Reasoning for Prognostic Learning in CBM Systems", *IEEE Aerospace Conference 2001 Proceedings*, vol. 1-7, pp. 2957-2969.

(Gebraeel et al, 2004) Gebraeel, N., Lawley, M., Liu, R., and Parmeshwaran, V. 2004, "Residual Life Predictions from Vibration-Based Degradation Signals: A Neural Network Approach", *IEEE Transactions on Industrial Electronics*, vol. 51, no. 3, pp. 694-700.

(Glorennec, 2000) P. Y. Glorennec, "Algorithmes d'apprentissage pour systèmes d'inférence floue" *HERMES, Sciences Publication*. 2000.

(Goddu et al, 1998) Goddu, G., Li, B., Chow, M.-Y., and Hung, J.C. 1998, "Motor Bearing Fault Diagnosis by A Fundamental Frequency Amplitude Based Fuzzy Decision System", *IEEE Annual Conference Proceedings of Industrial Electronics Society*, vol. 1-4, pp. 1961-1965.

(Goode et Chow, 1995) Goode, P.V., and Chow, M.-Y. 1995, "Using A Neural/Fuzzy System to Extract Heuristic Knowledge of Incipient Faults in Induction Motors: Part II Application", *IEEE Transactions on Industrial Electronics*, vol. 42, no. 2, pp. 139-146.

(Goumas et al., 2001) Goumas, S., Zervakis, M., Pouliezios, A., and Stavrakakis, G.S. 2001, "Intelligent On-line Quality Control of Washing Machines Using Discrete Wavelet Analysis Features and Likelihood Classification", *Engineering Applications of Artificial Intelligence*, vol. 14, no. 5, pp. 655-666.

(Gryllias et Antoniadis, 2012) Gryllias K.C., & Antoniadis I. A. (2012). " A Support Vector Machine approach based on physical model training for rolling element bearing fault detection in industrial environments ". *Engineering applications of Artificial Intelligence*, Vol 25, pp 326– 344.

(Harrouche et Felkaoui, 2014) Fateh Harrouche, Ahmed Felkaoui, (2014) «Automation of fault diagnosis of bearing by application of fuzzy inference system(FIS) » *Journal of Mechanics and Industry*. 2(2), 477-485(2014). DOI: 10.1051/meca/2014059.

(Haykin, 1999) Haykin S, 1999, "Neural Networks. A Comprehensive Foundation, Second Edition", *Pearson Prentice-Hall Publications, Ontario, Canada*.

(Heng et al., 2009) Heng, A., Zhang, S., Tan, A.C.C., and Mathew, J. 2009, "Rotating Machinery Prognostics: State of the Art, Challenges and Opportunities", *Mechanical Systems and Signal Processing*, vol. 23, no. 3, pp. 724-739.

(Hong et Chen, 1999) T.P. Hong, J.B. Chen. "Finding relevant attributes and membership functions". *Fuzzy Sets and Systems*, 104(1):389-404, 1999.

(Hu et al, 2003) Hu, N.-S., He, N.-N., and Hu, S. 2003, "Fault Diagnosis of the Steam Turbine Condenser System Based on SOM Neural Network", *International Conference on Machine Learning and Cybernetics*, vol. 1-5, pp. 1222-1225.

(Huguenin, 2004) Gérald Huguenin " logique floue et processeur flou " *EIAJ, CH-2610 Saint- Imier, courriel*. 24 Mars 2004

(Ilott et Griffiths,1997) Ilott, P.W., and Griffiths, A.J. 1997, "Fault Diagnosis of Pumping Machinery Using Artificial Neural Networks", *Proceedings of the Institution of Mechanical Engineers*, vol. 211, no. E3, pp. 185-194.

(Jack et Nandi, 2002); Jack L.B., & Nandi A. K. (2002). " Fault detection using support vector machines and artificial neural networks, augmented by genetic algorithms ". *Mechanical Systems and Signal Processing*, Vol 16, pp 373–390.

(Jang, 1993) Jang, J.-S.R. 1993, "ANFIS: Adaptive-Network-Based Fuzzy Inference System", *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 23 no. 3, pp. 665- 685.

(Jang et al, 1997) Jang, J.-S.R, Sun, C.-T., and Mizutani, E. 1997, "Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence", *Prentice-Hall, Upper Saddle River, New Jersey*.

(Khajavi et al, 2016) Mehrdad Nouri Khajavi, Ebrahim Nasernia et Mostafa Rostaghi, "Milling tool wear diagnosis by feed motor current signal using an artificial neural network", *Journal of Mechanical Science and Technology 30, KSME & Springer*, (11) (2016) 4869-4875

(Kohonen, 1997) Kohonen, T. 1997, "Self-Organizing Maps", *Second Edition*, Springer, New York.

(Konar et Chattopadhyay, 2011) Konar P., & Chattopadhyay P. (2011). " Bearing fault detection of induction motor using wavelet and Support Vector Machines (SVMs) ". *Applied Soft Computing*, Vol 11,pp 4203–4211.

(Kothamasu et Huang, 2007) Kothamasu, R., and Huang, S.H. 2007, "Adaptive Mamdani Fuzzy Model for Condition-based Maintenance", *Fuzzy Sets and Systems*, vol. 158, no. 24, pp. 2715-2733.

(Kowalski et Kowalska, 2003) Kowalski, C.T., and Orłowska-Kowalska, T. 2003, "Neural Networks Application for Induction Motor Faults Diagnosis", *Mathematics and Computers in Simulation*, vol. 63, no. 3-5, pp. 435-448.

(Kudo et al., 2000) M. Kudo, J. Sklansky, " Comparison of algorithms that select features for pattern classifiers", *Pattern Recognition*, Vol. 33, pp. 25-41, 2000.

(Kuo et al, 2002) Kuo, H.-C., Wu, L.-J., and Chen, J.-H. 2002, "Neural-fuzzy Fault Diagnosis in A Marine Propulsion Shaft System", *Journal of Materials Processing Technology*, vol. 122, no. 1, pp. 12-22.

- (Kurek et Osowski, 2010)** Kurek J., & Stokowski S. (2010). "Support vector machine for fault diagnosis of the broken rotor bars of squirrel-cage induction motor". *Neural Computing & Application*, Vol 19, pp 557–564.
- (Lei et al.,2008)** Lei, Y. G., He, Z. J., & Zi, Y. Y. (2008). "A new approach to intelligent fault diagnosis of rotating machinery". *Expert Systems with Applications*, 35(4), 1593–1600.
- (Leonhardt et Ayoubi, 1997)** Leonhardt, S., and Ayoubi, M. 1997, "Methods of Fault Diagnosis". *Control Engineering Practice*, vol. 5, no. 5, pp. 683-692.
- (Li et al. 2004)** Li, L.X., Mechefske, C.K., and Li, W.D. 2004, "Electric Motor Faults Diagnosis Using Artificial Neural Networks", *INSIGHT*, vol. 46, no. 10, pp. 616-621.
- (Li et al, 2009)** Li, H., Zhang, Y., and Zheng, H. 2009, "Gear Fault Detection and Diagnosis Under Speed-up Condition Based on Order Cepstrum and Radial Basis Function Neural Network", *Journal of Mechanical Science and Technology*, vol. 23, no. 10, pp. 2780-2789.
- (Li et Wu, 1989)** Li, C.J., and Wu, S.M. 1989, "On-line Detection of Localized Defects in Bearings by Pattern Recognition Analysis", *Transaction of ASME,Journal of Engineering for Industry*, vol. 101, no. 4, pp. 331-336.
- (Liao et al, 2005)** Liao, G., Shi, T., and Xuan, J. 2005, "Feature Selection and Condition Monitoring of Gearbox Using SOM", *IEEE International Joint Conference on Neural Networks (IJNN)*, vol. 1-5, pp. 2313-2318.
- (Liu et Mengel, 1992)** Liu, T.I., and Mengel, J.M. 1992, "Intelligent Monitoring of Ball Bearing Conditions", *Mechanical Systems and Signal Processing*, vol. 6, no. 5, pp. 419-431.
- (Liu et Shi, 2001)** Liu, S., and Shi, W. 2001, "Rough Set Based Intelligence Diagnostic System for Valves in Reciprocating Pumps", *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 1-5, pp. 353-358.
- (Loparo, 2013)** Loparo K. A. (2013). "Bearings Vibration Data Sets", *Case Western Reserve University*: <http://csegroups.case.edu/bearingdatacenter/home>.
- (Lou et Loparo, 2004)** Lou, X. S., & Loparo, K. A. (2004). "Bearing fault diagnosis based on wavelet transform and fuzzy inference". *Mechanical Systems and Signal Processing*, 18, 1077–1095.
- (Lu et al, 2011)** Lu, C., Ma, N., and Wang, Z. 2011, "Fault Detection for Hydraulic Pump Based on Chaotic Parallel RBF Network", *EURASIP Journal on Advances in Signal Processing*, vol. 49.

(Mahamad et Hiyama, 2011) Mahamad, A.K., and Hiyama, T. 2011, "Fault Classification Based Artificial Intelligent Methods of Induction Motor Bearing", *International Journal of Innovative Computing Information and Control*, vol. 7, no. 9, pp. 5477-5494.

(Malhi et al, 2011) Malhi, A., Yan, R., and Gao, R.X. 2011, "Prognosis of Defect Propagation Based on Recurrent Neural Networks", *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 703-711.

(Mathworks, 2000) Mathworks, "Fuzzy logic Toolbox-for Use MATLAB®", *User manual of Mathworks* (2000).

(McCormick et Nandi, 1997) McCormick, A.C., and Nandi, A.K. 1997, "Real-time Classification of Rotating Shaft Loading Conditions Using Artificial Neural Networks", *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 748-757.

(Mechefske, 1998) C. K. Mechefske, "Objective Machinery Fault Diagnosis Using Fuzzy Logic ", *Mechanical Systems and Signal Processig* (1998) 12(6), 885-862.

(Mechefske, 1992a) C. K. MECHEFSKE and J. Mathew, "Fault detection and diagnosis in low speed rolling element bearings, part I : the use of a parametric spectra", *Mechanical Systems and Signal Processing* 6(1992a), 297-307.

(Mechefske, 1992b) C. K. MECHEFSKE and J. Mathew, "Fault detection and diagnosis in low speed rolling element bearings, part I I: the use of nearset neighbour classification", *Mechanical Systems and Signal Processing* 6(1992b), 309-316.

(Meesad et Yen, 2000) Meesad, P., and Yen, G.G. 2000, "Pattern Classification by A Neurofuzzy Network: Application to Vibration Monitoring", *ISA Transactions*, vol. 39, no. 3, pp. 293-308.

(Meunie et Comarsal, 2003) B.B. Meunie et Comarsal "Logique floue : principe, aide à la d écision" *Edition Hermes science*, lavopsier 2003.

(Miguel et Blazquez, 2005) Miguel, L.J. d., and Blazquez, L.F. 2005, "Fuzzy Logic-based Decision-Making for Fault Diagnosis in A DC Motor", *Engineering Applications of Artificial Intelligence*, vol. 18, no. 4, pp. 423-450.

(Mohammadi et al, 2011) Mohammadi, R., Naderi, E., Khorasani, K., and Hashtrudi-Zad, S. 2011, "Fault Diagnosis of Gas Turbine Engines by Using Dynamic Neural Networks",*Proceedings of the IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS)*.

(Mokhtari, 1998) Mohand MOKHTARI, M.Marie "applications de Matlab®5 et Simulink®2", *Edition Springer France* 1998.

(Onel et Benbouzid, 2008) Onel, I.Y., and Benbouzid, M.E.H. 2008, "Induction Motors Bearing Failures Detection and Diagnosis Using a RBF ANN Park Pattern Based Method", *International Review of Electrical Engineering*, vol. 3, no. 1, pp. 159-165.

(Oukhellou, 1997) L. Oukhellou, "Paramétrisation et Classification de Signaux en Contrôle Non Destructif. Application à la Reconnaissance des Défauts de Rails par Courants de Foucault," *Thèse de l'Université de Paris-Sud*, 1997, pp 211.

(Patton et al, 2000) Patton, R.J., Chen, J., and Benkhedda, H. 2000, "A Study on Neuro-fuzzy Systems for Fault Diagnosis", *International Journal of Systems Science*, vol. 31, no. 11, pp. 1441-1448.

(Paya et al, 1997) Paya, B.A., and Esat, I.I. 1997, "Artificial Neural Network Based Fault Diagnostics of Rotating Machinery Using Wavelet Transforms as A Processor", *Mechanical Systems and Signal Processing*, vol. 11, no. 5, pp. 751-765.

(Peng, et al., 2010) Peng, Y., Dong, M., and Zuo, M.J. 2010, "Current Status of Machine Prognostics in Condition-based Maintenance: A Review", *The International Journal of Advanced Manufacturing Technology*, vol. 50, no. 1-4, pp. 297-313.

(Ping, 2000) Chen Ping, "bearing Condition Monitoring and Fault Diagnosis", *master of science thesis, the university of calgary, Canada*, 2000, pp.132.

(Premrudeepreechacharn et al, 2002) Premrudeepreechacharn, S., Utthiyoung, T., Kruepengkul, K., and Puongkaew, W. 2002, "Induction Motor Fault Detection and Diagnosis Using Supervised and Unsupervised Neural Networks", *IEEE International Conference on Industrial Technology*, vol. 1-2, pp. 93-96.

(Prieto et al, 2013) Prieto, M.D., Cirrincione, C., Espinosa, A.G., Ortega, J.A., and Henao, H. 2013, "Bearing Fault Detection by A Novel Condition Monitoring Scheme Based on Statistical-Time Features and Neural Networks", *IEEE Transactions on Industrial Electronics*, vol. 60, no. 8, pp. 3398-3407.

(Rafiee et al., 2007) Rafiee J., Arvani F., Harifi A., & Sadeghi M. H. (2007). " Intelligent condition monitoring of a gearbox using artificial neural network ". *Mechanical Systems and Signal Processing*, 21, pp1746–1754
on Condition Monitoring and Diagnostic Engineering (Comadem 2012), vol. 364.

(Rao, 2012) Rao, B.K.N, Pai, P.S, and Nagabhushana, T.N. 2012, "Failure Diagnosis and Prognosis of Rolling Element Bearings Using Artificial Neural Networks: A Critical Overview", *the 25th International Congress*

(Reddy et Mohanta, 2007) Reddy, M.J., and Mohanta, D.K. 2007, "A Wavelet-neuro-fuzzy Combined Approach for Digital Relaying of Transmission Line Faults ", *Electric Power Components and Systems*, vol. 35, no. 12, pp. 1385-1407.

(Rong et al, 2009) Rong, J., Zhang, X.-W., Chen, X.-Y., Li, H., Liu, J., and Song, X.-F. 2009, "Hydraulic Turbines Vibration Fault Diagnosis by RBF Neural Network Based On Particle Swarm Optimization", *Asia-Pacific Power and Energy Engineering Conference 2009*, vol. 1-7, pp. 961-964.

(Samanta et al., 2001) Samanta B., Al-Balushi K.R, & Al-Araimi S.A. (2001). "Use of genetic algorithm and artificial neural network for gear condition diagnostics ". *Proceedings of COMADEM*, University of Manchester, UK, pp 449–456.

(Samanta et Al- Balushi, 2003) Samanta, B., and Al-Balushi, K.R. 2003, "Artificial Neural Network Based Fault Diagnostics of Rolling Element Bearings Using Time-Domain Feature", *Mechanical Systems and Signal Processing*, vol. 17, no. 2, pp. 317-328.

(Saravanan et al, 2009) Saravanan, N., Cholairajan, S., and Ramachandran, K.I. 2009, "Vibration-based Fault Diagnosis of Spur Bevel Gear Box Using Fuzzy Technique", *Expert Systems with Applications*, vol. 36, no.2, pp. 3119-3135.

(Satish et Sarma, 2005) Satish, B., and Sarma, N.D.R. 2005, "A Fuzzy BP Approach for Diagnosis and Prognosis of Bearing Faults in Induction Motors", *IEEE Power Engineering Society General Meeting*, pp. 2291-2294

(Selaimia, 2006) Selaimia, Y., Moussaoui, A., and Abbassi, H.A. 2006, "Multi Neural Networks Based Approach for Fault Detection and Diagnosis of A DC-Motor", *Neural Network World*, vol. 16, no. 5, pp. 369-379.

(Senguler et Al, 2010) Senguler, T., Karatoprak, E., and Seker, S. 2010, "A New MLP Approach for the Detection of the Incipient Bearing Damage", *Advances In Electrical and Computer Engineering*, vol. 10, no. 3, pp. 34-39.

(Shen et al, 2013) Shen C., Wang D., Kong F., & Tse P. W. (2013). "Fault diagnosis of rotating machinery based on the statistical parameters of wavelet packet paving and a generic support vector regressive classifier". *Measurement*, Vol 46, pp 1551–1564.

(Sreejith et al, 2008) Sreejith, B., Verma, A.K., and Srividya A. 2008, "Fault Diagnosis of Rolling Element Bearing Using Time-domain Features and Neural Networks", *IEEE Region 10 Colloquium and the Third International Conference on Industrial and Information Systems*, vol. 1-2, pp. 619-624.

(Sugeno et Kang, 1988) Sugeno, M., and Kang, G.T. 1988, "Structure Identification of Fuzzy Model", *Fuzzy Sets and Systems*, vo. 28, no. 1, pp. 15-33.

(Sugumaran et al, 2007) V. Sugumaran; K.I. Ramachandran, "Automatic rule learning using decision tree for fuzzy classifier in fault diagnosis of roller bearing", *Mechanical Systems and Signal Processing*, Vol 21 (2007) 2237-2247.

(Sun, 1998) Q. SUN, F.XI, and G. Krishnappa, "signature analysis of rolling element defects", *Proceeding of CMSE Forum, Toronto* (1998), pp.423-429.

(Takagi et Sugeno, 1985) Takagi, T., and Sugeno, M. 1985, "Fuzzy Identification of Systems and Its Applications to Modelling and Control", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, no. 1, pp. 116-132.

(Tsoukalas et Uhrig, 1997) Tsoukalas, L.H., and Uhrig, R.E. 1997, "Fuzzy and Neural Approaches in Engineering", John Wiley & Sons, New York.

(Umbrajkaar et Krishnamoorthy, 2018) Amit Umbrajkaar , Arunagiri Krishnamoorthy, "Vibration analysis using wavelet transform and fuzzy logic for shaft misalignment", *Journal of Vibroengineering(JVE)*, Vol. 20, Issue 8, 2018, p. 2855-2865

(Vicente et al, 2001) Vicente, S.A.d., Fujimoto, R.Y., and Padovese, L.R. 2001, "Rolling Bearing Fault Diagnostic System Using Fuzzy Logic", *the 10th International Conference on Fuzzy System*, vol. 1-3, pp. 816-819.

(Wang et al, 2004a) Wang, W., Ismail, F., and Golnaraghi, F. 2004a, "A Neuro-Fuzzy Approach to Gear System Monitoring", *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 5, pp. 710-723.

(Wang et al, 2004b) Wang, W.Q., Golnaraghi, M.F., and Ismail, F. 2004b, "Prognosis of Machine Health Condition Using Neuro-fuzzy Systems", *Mechanical Systems and Signal Processing*, vol. 18, no. 4, pp. 813-831.

(Wang et al, 2010) Wang, C.-C., Kang, Y., Shen, P.-C., Chang, Y.-P., and Chung, Y.-L. 2010, "Applications of Fault Diagnosis in Rotating Machinery by Using Time Series Analysis with Neural Network", *Expert Systems with Applications*, vol. 37, no. 2, pp. 1696-1702.

(Wang et Lei, 2001) Wang, K., and Lei, B. 2001, " Using B-spline Neural Network to Extract Fuzzy Rules for A Centrifugal Pump Monitoring ", *Journal of Intelligent Manufacturing*, vol. 12, no. 1, pp. 5-11.

(Wang et Too, 2002) Wang C.C, & TOO G.P.J. (2002). " Rotating machine fault detection based on HOS and artificial neural networks ". *Journal of intelligent manufacturing*, Vol 13, 283-293.

(Widron et Lehr, 1960) Widrow, B., and Lehr, M.A. 1960, "Adaptive Switching Circuits", *IRE Western Electric Show and Convention Record*, part 4, pp. 96-104.

(Widron et Lehr, 1990) Widrow, B., and Lehr, M.A. 1990, "30 Years of Adaptive Neural Networks: Perceptron, Madaline, & Backpropagation", *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1415-1442.

(Worden et al, 2011) Worden K, Staszewski WJ, & Hensman J.J. (2011). " *Natural computing for mechanical systems research: A tutorial overview* ", *Mechanical Systems and Signal Processing*, Vol 25 (2011), pp 4–111.

(Wu et al, 2002) Wu, J.-F., Hu, N.-S., Hu, S., and Zhao, Y. 2002, "Application of SOM Neural Network in Fault Diagnosis of the Steam Turbine Regenerative System", *International Conference on Machine Learning and Cybernetics*, vol. 1-4, pp. 184-187.

(Wu et Chan, 2009) Wu, J.-D., and Chan, J.-J. 2009, "Faulted Gear Identification of a Rotating Machinery Based on Wavelet Transform and Artificial Neural Network", *Expert Systems with Applications*, vol. 36, no. 5, pp. 8862-8875.

(Wu et Hsu, 2009) Wu, J.-D., and Hsu, C.-C. 2009, "Fault Gear Identification Using Vibration Signal with Discrete Wavelet Transform Technique and Fuzzy–logic Inference", *Expert Systems with Applications*, vol. 36, no. 2, pp. 3785-3794.

(Yam et al, 2001), Yam, R.C.M., Tse, P.W., Li, L., and Tu, P. 2001, "Intelligent Predictive Decision Support System for Condition-based Maintenance", *Advanced Manufacturing Technology*, vol. 17, no. 5, pp. 383-391.

(Yang et al, 2002) Yang, H., Mathew, J., and Ma, L. 2002, "Intelligent Diagnosis of Rotating Machinery Faults - A Review", *the 3rd Asia-Pacific Conference on Systems Integrity and Maintenance, Cairns, Australia*.

(Yang et al, 2004) Yang, B.S., Han, T., and An, J.L. 2004, "ART-Kohonen Neural Network for Fault Diagnosis of Rotating Machinery", *Mechanical Systems and Signal Processing*, vol. 18, no. 3, pp. 645-657.

(Yang et al, 2008) Yang, S., Li, W., and Wang, C. 2008, "The Intelligent Fault Diagnosis of Wind Turbine Gearbox Based on Artificial Neural Network", *Proceedings of International Conference on Condition Monitoring and Diagnosis*, pp. 1327- 1330.

(Zadeh, 1965) L.A. Zadeh, "Fuzzy Sets, Information and Control", Vol. 8, PP 338-352, 1965.

(Zadeh, 1971) L, Zadeh. "Quantitative fuzzy semantics. *Information science*", 3(1) ; 159-176,1971.

(Zadeh, 1975) L. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning", parts 1 and 2, 1975, pp 199–249 and 301–357.

(Zadeh, 1976) L.A.Zadeh, "A fuzzy algorithmic approach to the definition of complex and imprecise concepts", *Int.j. Man-machine Studies*, pp 249-291, 1976.

(Zadeh, 1983) L. Zadeh, "The role of fuzzy logic in the management of uncertainty in expert systems", *Fuzzy Sets and Systems*, 1983, p. 199–227.

(Zarei et al, 2008) Zarei, J., Poshtan, J., and Poshtan, M. 2008, "Bearing Fault Detection in Induction Motor Using Pattern Recognition Techniques", *IEEE 2nd International Power and Energy Conference*, vol. 1-3, pp. 749-753.

(Zhang et al., 2010) Long Zhang , Guoliang Xiong , Hesheng Liu, Huijun Zou , Weizhong Guo . (2010). "Bearing fault diagnosis using multi-scale entropy and adaptive neuro-fuzzy inference". *Expert Systems with Applications*, 37 (2010), 6077–6085.

(Zhang et al, 2013) Zhang Y., Zuo H., et Bai F. (2013). "Classification of fault location and performance degradation of a roller bearing". *Measurement*, Vol 46, pp 1178–1189.

(Zhang et Ganesan, 1997) Zhang, S., and Ganesan, R. 1997, "Multivariable Trend Analysis Using Neural Networks for Intelligent Diagnostics of Rotating Machinery", *Journal of Engineering for Gas Turbines and Power - Transactions of the ASME*, vol. 119, no. 2, pp. 378-384.

(Zhang et Morris, 1996) Zhang, J., and Morris, J. 1996, "Process Modelling and Fault Diagnosis Using Fuzzy Neural Networks", *Fuzzy Sets and Systems*, vol. 79, no. 1, pp. 127-140.

(Zhao et al, 2009) Zhao, F., Cheng, J., Guo, L., and Li, X. 2009, "Neuro-fuzzy Based Condition Prediction of Bearing Health", *Journal of Vibration and Control*, vol. 15, no. 7, pp. 1079-1091.

(Zhong et al. 2005) Zhong, F., Shi, T., and He, T. 2005, "Fault Diagnosis of Motor Bearing Using Self- Organizing Maps", *Proceedings of the Eighth International Conference on Electrical Machines and Systems*, vol. 1-3 pp. 2411-2414.

ANNEXES

ANNEXE (4.3)

**Matrice des 11 indicateurs pour les 28 tranches
du signal (sans défaut) et avec défaut (0.007inch).**

Remarque : pour les deux tableaux 4.3.1 et 4.3.2 on a :

En lignes : les tranches obtenus par la division du signal de 1 à 28.

En colonnes : les indicateurs calculés de 1 à 11, et qui sont ; respectivement ET, F_c, A_{cc}, K_s, K_u, E_{BT}, E_{B1}, E_{B2}, E_{B3}, E_{B4}, E_{B5}.

Tableau 4. 3.1 – matrice des 11 indicateurs pour les 28 tranches du signal (sans défaut).

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|----|----------|--------|---------|-------------|--------|--------------|---------|---------|---------|---------|--------|
| 1 | 0.030957 | 3.2566 | 0.19069 | -0.01665 | 2.8269 | -1.3323e+005 | -8541.3 | -8764.7 | -9032.4 | -9308 | -10767 |
| 2 | 0.031044 | 3.5109 | 0.2037 | 0.00035599 | 2.7847 | -1.3302e+005 | -8508.5 | -8671.6 | -9007.8 | -9297.5 | -10778 |
| 3 | 0.031006 | 3.8276 | 0.20338 | 0.0046177 | 2.8565 | -1.3263e+005 | -8527 | -8627.6 | -9000.6 | -9305.5 | -10785 |
| 4 | 0.030798 | 3.7729 | 0.21194 | -0.025439 | 2.9176 | -1.33e+005 | -8516.7 | -8613.8 | -8994.3 | -9325.2 | -10744 |
| 5 | 0.031367 | 3.8035 | 0.2132 | -0.048633 | 3.0986 | -1.323e+005 | -8458.4 | -8600 | -8934 | -9311.2 | -10725 |
| 6 | 0.031065 | 4.2071 | 0.219 | -0.028562 | 3.2119 | -1.3257e+005 | -8494.8 | -8642.2 | -8994.1 | -9348.7 | -10749 |
| 7 | 0.030597 | 3.837 | 0.2054 | -0.0035019 | 2.8706 | -1.3327e+005 | -8575.8 | -8690.8 | -9022.7 | -9355.9 | -10775 |
| 8 | 0.031342 | 3.5703 | 0.2306 | -0.014419 | 2.8901 | -1.3257e+005 | -8471.1 | -8600.9 | -8998.5 | -9320 | -10803 |
| 9 | 0.031722 | 3.6375 | 0.23167 | -0.011709 | 3.0041 | -1.324e+005 | -8387.4 | -8570 | -8972 | -9310 | -10793 |
| 10 | 0.031145 | 4.1436 | 0.21308 | -0.035041 | 3.0593 | -1.329e+005 | -8458.1 | -8689.5 | -8982.9 | -9343.3 | -10774 |
| 11 | 0.031168 | 3.7193 | 0.2112 | -0.045249 | 2.9514 | -1.3338e+005 | -8489.6 | -8667.7 | -8976.1 | -9320.4 | -10796 |
| 12 | 0.03057 | 3.4267 | 0.20528 | -0.028437 | 2.8852 | -1.3317e+005 | -8495.1 | -8648.5 | -8986.8 | -9311.5 | -10785 |
| 13 | 0.030119 | 3.3094 | 0.20316 | -0.024947 | 2.8535 | -1.3361e+005 | -8586.7 | -8743.8 | -9063.8 | -9351.9 | -10841 |
| 14 | 0.031153 | 3.5077 | 0.19621 | -0.016166 | 2.8711 | -1.3316e+005 | -8520.7 | -8671.7 | -8991 | -9303.3 | -10793 |
| 15 | 0.031349 | 3.5513 | 0.20794 | 0.00025797 | 2.8934 | -1.3293e+005 | -8539 | -8657.9 | -8977.5 | -9330.7 | -10726 |
| 16 | 0.031043 | 3.5381 | 0.20678 | -0.0089068 | 2.8831 | -1.3295e+005 | -8568 | -8661.8 | -8993.7 | -9348.3 | -10764 |
| 17 | 0.0311 | 3.4771 | 0.19916 | -0.016952 | 2.8268 | -1.3333e+005 | -8630.1 | -8633.9 | -9027.6 | -9328.8 | -10758 |
| 18 | 0.030964 | 3.5872 | 0.20573 | -0.0084451 | 2.8169 | -1.3315e+005 | -8541.9 | -8650.4 | -8994.5 | -9321.1 | -10784 |
| 19 | 0.030642 | 3.4983 | 0.2033 | -0.01228 | 2.8225 | -1.3274e+005 | -8553.4 | -8738 | -8998.8 | -9294.8 | -10801 |
| 20 | 0.031158 | 3.4325 | 0.20775 | -0.010471 | 2.8361 | -1.3319e+005 | -8470.5 | -8689.3 | -8948.5 | -9329.8 | -10788 |
| 21 | 0.031499 | 4.9532 | 0.21575 | -0.00028631 | 3.3266 | -1.3258e+005 | -8483.5 | -8625.9 | -8976.5 | -9313.2 | -10777 |
| 22 | 0.030855 | 3.8135 | 0.21309 | -0.029545 | 2.9604 | -1.3313e+005 | -8538.8 | -8656.1 | -9008.9 | -9382.4 | -10777 |
| 23 | 0.031034 | 3.4383 | 0.19394 | -0.036363 | 2.7992 | -1.3251e+005 | -8509.8 | -8729.8 | -8965.7 | -9378.1 | -10756 |
| 24 | 0.031144 | 3.4994 | 0.19991 | -0.03063 | 2.8696 | -1.3295e+005 | -8488.2 | -8656.6 | -9001.2 | -9363.6 | -10829 |
| 25 | 0.031118 | 3.9521 | 0.20805 | -0.012897 | 2.9889 | -1.3321e+005 | -8493.8 | -8635.4 | -8942.6 | -9318.7 | -10817 |
| 26 | 0.031505 | 4.8908 | 0.21904 | -0.0049052 | 3.3431 | -1.3289e+005 | -8437.8 | -8580.7 | -8919.8 | -9262 | -10735 |
| 27 | 0.031138 | 4.6204 | 0.21405 | -0.025611 | 3.3488 | -1.3252e+005 | -8486.7 | -8619.6 | -8946.6 | -9355.5 | -10754 |
| 28 | 0.030622 | 6.2096 | 0.21353 | 0.0045282 | 4.784 | -1.324e+005 | -8599.1 | -8690.7 | -9012.9 | -9345.4 | -10764 |

Tableau 4. 3.2 – matrice des 11 indicateurs pour les 28 tranches du signal du roulement avec défaut (diamètre (0.007inch)).

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|----|---------|--------|--------|------------|--------|--------|---------|---------|---------|---------|---------|
| 1 | 0.16645 | 4.5867 | 1.5349 | -0.0092705 | 4.519 | -93687 | -5153.1 | -5746.8 | -5750.4 | -6353.4 | -5825.2 |
| 2 | 0.16251 | 4.5268 | 1.4311 | -0.0080025 | 4.5388 | -92397 | -5287.7 | -5924.3 | -5852.3 | -6479.5 | -5993.3 |
| 3 | 0.16114 | 4.4029 | 1.3838 | -0.0097401 | 4.5837 | -93715 | -5315.8 | -6007.5 | -5846.1 | -6463.7 | -6029.1 |
| 4 | 0.1614 | 4.8301 | 1.405 | -0.0080857 | 4.795 | -93433 | -5274.1 | -5916.3 | -5874.8 | -6470.4 | -6014.6 |
| 5 | 0.16201 | 4.3806 | 1.4026 | -0.0019278 | 4.5768 | -91932 | -5244.8 | -5838.1 | -5854.7 | -6439.4 | -5965.2 |
| 6 | 0.16392 | 4.4167 | 1.4373 | -0.0021543 | 4.5348 | -94095 | -5213.8 | -5789.4 | -5742.5 | -6366.7 | -5904.9 |
| 7 | 0.16665 | 4.3239 | 1.4371 | -0.0054867 | 4.4883 | -90757 | -5187.3 | -5750 | -5704.2 | -6306.1 | -5820.6 |
| 8 | 0.16675 | 4.3222 | 1.4589 | -0.0067694 | 4.475 | -91285 | -5170.1 | -5746.1 | -5696.4 | -6334.7 | -5791.2 |
| 9 | 0.16663 | 4.3783 | 1.4659 | 0.0003266 | 4.4861 | -92580 | -5183.1 | -5765.9 | -5688 | -6346.8 | -5845.8 |
| 10 | 0.16563 | 4.3621 | 1.4321 | -0.0057054 | 4.4774 | -92715 | -5250.8 | -5848.6 | -5731 | -6348.3 | -5882.5 |
| 11 | 0.16366 | 4.4137 | 1.4353 | -0.0095217 | 4.5129 | -92733 | -5263.2 | -5848.9 | -5778.1 | -6365.8 | -5902.1 |
| 12 | 0.1641 | 4.519 | 1.4495 | -0.0092049 | 4.5735 | -93448 | -5235.9 | -5825.1 | -5778.4 | -6414.4 | -5941.7 |
| 13 | 0.16433 | 4.6574 | 1.4523 | -0.01028 | 4.6887 | -91865 | -5250.6 | -5887.1 | -5779.5 | -6406.4 | -5960.8 |
| 14 | 0.16557 | 4.64 | 1.4825 | -0.000956 | 4.6711 | -92173 | -5168 | -5857.5 | -5782.7 | -6359.3 | -5906.2 |
| 15 | 0.16721 | 4.5186 | 1.4717 | -0.0049412 | 4.617 | -90232 | -5183.8 | -5797.3 | -5697 | -6346 | -5840.6 |
| 16 | 0.1661 | 4.566 | 1.4695 | -0.0065709 | 4.6532 | -90970 | -5090.3 | -5817.6 | -5721.5 | -6319.7 | -5878.2 |
| 17 | 0.16393 | 4.5999 | 1.4529 | 0.0048165 | 4.6731 | -92126 | -5177.3 | -5896.1 | -5888.3 | -6401.1 | -6001.2 |
| 18 | 0.16336 | 4.5403 | 1.4317 | 0.0058761 | 4.666 | -91483 | -5228.3 | -5814.7 | -5888 | -6430.6 | -5998.9 |
| 19 | 0.16448 | 4.5922 | 1.4741 | 0.0057128 | 4.6865 | -90553 | -5185.3 | -5807.7 | -5735.8 | -6367.1 | -5932.7 |
| 20 | 0.1659 | 4.4845 | 1.447 | 0.0026887 | 4.6588 | -91855 | -5136.9 | -5788.1 | -5741 | -6353.3 | -5987.4 |
| 21 | 0.16747 | 5.2683 | 1.4867 | -0.004002 | 4.962 | -92079 | -5114.6 | -5718.4 | -5715.7 | -6304.7 | -5864.1 |
| 22 | 0.17026 | 4.4653 | 1.5609 | -0.0071344 | 4.6521 | -90798 | -5052.7 | -5653.4 | -5631.7 | -6281.7 | -5814.3 |
| 23 | 0.17001 | 4.3757 | 1.5515 | -0.013806 | 4.7079 | -91319 | -4969.5 | -5645.8 | -5691.3 | -6310.8 | -5792 |
| 24 | 0.1656 | 4.4906 | 1.5025 | -0.0090402 | 4.6717 | -93486 | -5117.4 | -5799 | -5804.9 | -6411.9 | -5910.5 |
| 25 | 0.16366 | 4.3626 | 1.428 | -0.0031413 | 4.6019 | -93512 | -5239.9 | -5893.3 | -5842.9 | -6416.4 | -6041.3 |
| 26 | 0.1686 | 4.5515 | 1.5702 | -0.0065552 | 4.6478 | -92146 | -5087.2 | -5731.1 | -5703.5 | -6321.2 | -5839.4 |
| 27 | 0.17096 | 5.0472 | 1.5909 | -0.004935 | 4.8301 | -91898 | -5016.3 | -5679.6 | -5688 | -6328.3 | -5776.4 |
| 28 | 0.16519 | 4.9027 | 1.455 | -0.0064024 | 4.8177 | -92358 | -5119.2 | -5789.8 | -5788 | -6393.6 | -5902.6 |

ANNEXE (4.4)

- Base de données de roulements pour le système initial (FIS) de la première étape-

Remarque : cette annexe présente une matrice qui contient les valeurs numériques des indicateurs de la base de données des roulements selon les conditions de la première étape de classification citées dans la section (4.2.3.1). Donc on a :

En lignes : Etiquette d'observation (de 1 à 112).

En colonnes : Indicateurs (de 1 à 3, qui sont respectivement F_c , K_s , K_u).

| | 1 | 2 | 3 | | | | | | | | |
|----|--------|-------------|--------|-----|--------|---------|--------|-----|--------|--------|--------|
| 1 | 3.2566 | -0.0167 | 2.8269 | 51 | 4.3757 | -0.0138 | 4.7079 | 101 | 4.4295 | 0.0568 | 3.4840 |
| 2 | 3.5109 | 3.5599e-04 | 2.7847 | 52 | 4.4906 | -0.0090 | 4.6717 | 102 | 4.2738 | 0.0423 | 3.5713 |
| 3 | 3.8276 | 0.0046 | 2.8565 | 53 | 4.3626 | -0.0031 | 4.6019 | 103 | 4.5098 | 0.0360 | 3.4417 |
| 4 | 3.7729 | -0.0254 | 2.9176 | 54 | 4.5515 | -0.0067 | 4.6478 | 104 | 4.4889 | 0.0483 | 3.4478 |
| 5 | 3.8035 | -0.0486 | 3.0986 | 55 | 5.0472 | -0.0049 | 4.8301 | 105 | 4.5040 | 0.0462 | 3.5743 |
| 6 | 4.2071 | -0.0286 | 3.2119 | 56 | 4.9027 | -0.0064 | 4.8177 | 106 | 4.4492 | 0.0462 | 3.4081 |
| 7 | 3.8370 | -0.0035 | 2.8706 | 57 | 5.4580 | 0.0963 | 6.2342 | 107 | 4.1295 | 0.0369 | 3.1920 |
| 8 | 3.5703 | -0.0144 | 2.8901 | 58 | 5.2821 | 0.0961 | 6.1750 | 108 | 4.8529 | 0.0296 | 3.7460 |
| 9 | 3.6375 | -0.0117 | 3.0041 | 59 | 5.3845 | 0.0934 | 6.3004 | 109 | 4.4020 | 0.0464 | 3.4268 |
| 10 | 4.1436 | -0.0350 | 3.0593 | 60 | 5.3048 | 0.0834 | 6.2466 | 110 | 4.0549 | 0.0435 | 3.2275 |
| 11 | 3.7193 | -0.0452 | 2.9514 | 61 | 5.4832 | 0.0758 | 6.3647 | 111 | 4.8457 | 0.0328 | 3.5445 |
| 12 | 3.4267 | -0.0284 | 2.8852 | 62 | 5.7164 | 0.0688 | 6.8715 | 112 | 5.0192 | 0.0261 | 3.5518 |
| 13 | 3.3094 | -0.0249 | 2.8535 | 63 | 5.8085 | 0.0567 | 6.7074 | | | | |
| 14 | 3.5077 | -0.0162 | 2.8711 | 64 | 5.6193 | 0.0389 | 6.3290 | | | | |
| 15 | 3.5513 | 2.5797e-04 | 2.8934 | 65 | 5.6532 | 0.0251 | 6.1281 | | | | |
| 16 | 3.5381 | -0.0089 | 2.8831 | 66 | 5.6859 | 0.0225 | 6.2371 | | | | |
| 17 | 3.4771 | -0.0170 | 2.8268 | 67 | 5.4716 | 0.0240 | 6.1065 | | | | |
| 18 | 3.5872 | -0.0084 | 2.8169 | 68 | 5.1407 | 0.0253 | 5.7305 | | | | |
| 19 | 3.4983 | -0.0123 | 2.8225 | 69 | 5.1011 | 0.0375 | 5.7601 | | | | |
| 20 | 3.4325 | -0.0105 | 2.8361 | 70 | 5.3219 | 0.0532 | 6.3773 | | | | |
| 21 | 4.9532 | -2.8631e-04 | 3.3266 | 71 | 5.3771 | 0.0588 | 6.4035 | | | | |
| 22 | 3.8135 | -0.0295 | 2.9604 | 72 | 6.1087 | 0.0667 | 6.3999 | | | | |
| 23 | 3.4383 | -0.0364 | 2.7992 | 73 | 6.7603 | 0.0738 | 6.4966 | | | | |
| 24 | 3.4994 | -0.0306 | 2.8696 | 74 | 5.6261 | 0.0790 | 6.1782 | | | | |
| 25 | 3.9521 | -0.0129 | 2.9889 | 75 | 5.4500 | 0.0888 | 6.1684 | | | | |
| 26 | 4.8908 | -0.0049 | 3.3431 | 76 | 5.4080 | 0.0928 | 5.9299 | | | | |
| 27 | 4.6204 | -0.0256 | 3.3488 | 77 | 5.5813 | 0.0947 | 6.4233 | | | | |
| 28 | 6.2096 | 0.0045 | 4.7840 | 78 | 5.4269 | 0.0857 | 6.4174 | | | | |
| 29 | 4.5867 | -0.0093 | 4.5190 | 79 | 5.1636 | 0.0816 | 5.9176 | | | | |
| 30 | 4.5268 | -0.0080 | 4.5388 | 80 | 5.2491 | 0.0719 | 5.8256 | | | | |
| 31 | 4.4029 | -0.0097 | 4.5837 | 81 | 5.1350 | 0.0540 | 5.9045 | | | | |
| 32 | 4.8301 | -8.0570e-04 | 4.7950 | 82 | 5.6939 | 0.0497 | 6.2491 | | | | |
| 33 | 4.3806 | -0.0019 | 4.5768 | 83 | 6.0242 | 0.0499 | 6.7221 | | | | |
| 34 | 4.4167 | -0.0022 | 4.5348 | 84 | 6.0126 | 0.0390 | 6.9045 | | | | |
| 35 | 4.3239 | -0.0055 | 4.4883 | 85 | 5.0600 | 0.0558 | 3.7778 | | | | |
| 36 | 4.3222 | -0.0068 | 4.4750 | 86 | 4.0384 | 0.0384 | 3.2244 | | | | |
| 37 | 4.3783 | 3.2660e-04 | 4.4961 | 87 | 4.0356 | 0.0364 | 3.1782 | | | | |
| 38 | 4.3621 | -0.0057 | 4.4774 | 88 | 4.2989 | 0.0440 | 3.3743 | | | | |
| 39 | 4.4137 | -0.0095 | 4.5129 | 89 | 4.0969 | 0.0453 | 3.3864 | | | | |
| 40 | 4.5190 | -0.0092 | 4.5735 | 90 | 4.1376 | 0.0492 | 3.2490 | | | | |
| 41 | 4.6574 | -0.0103 | 4.6687 | 91 | 5.4115 | 0.0470 | 4.0182 | | | | |
| 42 | 4.6400 | -9.5600e-04 | 4.6711 | 92 | 4.7564 | 0.0561 | 3.5734 | | | | |
| 43 | 4.5186 | -0.0049 | 4.6170 | 93 | 6.5057 | 0.0632 | 7.9580 | | | | |
| 44 | 4.5660 | -0.0066 | 4.6532 | 94 | 4.3851 | 0.0476 | 3.4096 | | | | |
| 45 | 4.5999 | 0.0048 | 4.6731 | 95 | 4.1305 | 0.0464 | 3.3884 | | | | |
| 46 | 4.5403 | 0.0059 | 4.6660 | 96 | 4.4140 | 0.0412 | 3.4328 | | | | |
| 47 | 4.5922 | 0.0057 | 4.6065 | 97 | 4.6613 | 0.0276 | 3.3301 | | | | |
| 48 | 4.4845 | 0.0027 | 4.6588 | 98 | 4.2632 | 0.0506 | 3.2375 | | | | |
| 49 | 5.2683 | -0.0040 | 4.9620 | 99 | 4.2513 | 0.0510 | 3.2400 | | | | |
| 50 | 4.4653 | -0.0071 | 4.6521 | 100 | 6.5317 | 0.0464 | 8.1944 | | | | |

ANNEXE (4.6)

Résultat de classification de la base de données de roulements par le classifieur initial FIS de la première étape (avant l'optimisation)

Remarque :

Pour Cette annexe, on a :

En ligne (une seule ligne) : sortie du système FIS (état)

En colonne (de 1à112) : ces colonnes portent des valeurs réelles de la sortie du système FIS (résultats de classification des observations de la base de données).

| | | | | | | | | | | | | | | | | |
|---|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 1 | 3.9522 | 3.7542 | 4.0900 | 4.1296 | 4.2034 | 4.2860 | 3.8441 | 3.9715 | 3.9087 | 4.1999 | 4.1058 | 4.0113 | 3.9910 | 3.9970 | 3.7252 | 3.7570 |
| | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| 1 | 4.0218 | 3.7918 | 3.8953 | 3.7874 | 4.8228 | 4.1448 | 4.0607 | 4.0394 | 4.0949 | 4.7552 | 4.2163 | 5.3680 | 4.8235 | 4.7978 | 4.6264 | 5.2067 |
| | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 |
| 1 | 4.7748 | 4.8607 | 4.9530 | 4.9828 | 4.9446 | 4.9921 | 4.8003 | 4.6857 | 4.6532 | 4.8213 | 4.7597 | 4.7412 | 4.6828 | 4.6386 | 4.6628 | 4.6874 |
| | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 |
| 1 | 5.8970 | 4.6955 | 4.4808 | 4.6467 | 4.7437 | 4.7324 | 5.4950 | 5.3224 | 6.0693 | 6.1092 | 6.0723 | 6.0911 | 6.0580 | 6.1033 | 6.1994 | 6.3704 |
| | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 |
| 1 | 6.1298 | 5.9693 | 6.1329 | 6.7463 | 7.0419 | 6.3920 | 6.2227 | 6.0540 | 6.1073 | 6.0588 | 6.0804 | 6.1809 | 6.0470 | 6.0618 | 6.2941 | 6.3651 |
| | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 |
| 1 | 6.6686 | 6.3119 | 6.3067 | 6.4787 | 7.0275 | 5.8137 | 5.7739 | 5.9987 | 6.0201 | 5.8744 | 7.2737 | 6.4302 | 6.7195 | 6.0185 | 6.0322 | 6.0433 |
| | 97 | 98 | 99 | 100 | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 | 111 | 112 |
| 1 | 5.9317 | 5.8572 | 5.8624 | 7.3827 | 6.1133 | 6.2906 | 6.0484 | 6.0579 | 6.2482 | 6.0037 | 5.8216 | 6.8054 | 6.0380 | 5.8258 | 6.5375 | 6.6394 |

ANNEXE (4.8)

- Base de données de roulements pour la deuxième étape (optimisation structurelle) -

Remarque : cette annexe présente une matrice qui contient les valeurs numériques des indicateurs de la base de données des roulements selon les conditions citées dans la section (section 4.2.3.2.1). Donc on a :

En lignes : Etiquette d'observation (de 1 à 112).

En colonnes : valeurs des Indicateurs (de 1 à 3, qui sont respectivement ET, E_{B4} , E_{B5}).

| | 1 | 2 | 3 | | 1 | 2 | 3 |
|----|-------------|--------|-------------|----|-------------|--------|-------------|
| 1 | -1.0767e+04 | 0.0310 | -9.3080e+03 | 27 | -1.0754e+04 | 0.0311 | -9.3555e+03 |
| 2 | -1.0778e+04 | 0.0310 | -9.2975e+03 | 28 | -1.0764e+04 | 0.0306 | -9.3454e+03 |
| 3 | -1.0785e+04 | 0.0310 | -9.3055e+03 | 29 | -5.8252e+03 | 0.1665 | -6.3534e+03 |
| 4 | -1.0744e+04 | 0.0308 | -9.3252e+03 | 30 | -5.9933e+03 | 0.1625 | -6.4795e+03 |
| 5 | -1.0725e+04 | 0.0314 | -9.3112e+03 | 31 | -6.0291e+03 | 0.1611 | -6.4637e+03 |
| 6 | -1.0749e+04 | 0.0311 | -9.3487e+03 | 32 | -6.0146e+03 | 0.1614 | -6.4704e+03 |
| 7 | -1.0775e+04 | 0.0306 | -9.3559e+03 | 33 | -5.9652e+03 | 0.1620 | -6.4394e+03 |
| 8 | -1.0803e+04 | 0.0313 | -9.3200e+03 | 34 | -5.9049e+03 | 0.1639 | -6.3667e+03 |
| 9 | -1.0793e+04 | 0.0317 | -9.3100e+03 | 35 | -5.8206e+03 | 0.1667 | -6.3061e+03 |
| 10 | -1.0774e+04 | 0.0311 | -9.3433e+03 | 36 | -5.7912e+03 | 0.1667 | -6.3343e+03 |
| 11 | -1.0796e+04 | 0.0312 | -9.3204e+03 | 37 | -5.8458e+03 | 0.1666 | -6.3468e+03 |
| 12 | -1.0785e+04 | 0.0306 | -9.3115e+03 | 38 | -5.8825e+03 | 0.1656 | -6.3493e+03 |
| 13 | -1.0841e+04 | 0.0301 | -9.3519e+03 | 39 | -5.9021e+03 | 0.1637 | -6.3658e+03 |
| 14 | -1.0793e+04 | 0.0312 | -9.3033e+03 | 40 | -5.9417e+03 | 0.1641 | -6.4144e+03 |
| 15 | -1.0726e+04 | 0.0313 | -9.3307e+03 | 41 | -5.9608e+03 | 0.1643 | -6.4064e+03 |
| 16 | -1.0764e+04 | 0.0310 | -9.3483e+03 | 42 | -5.9062e+03 | 0.1656 | -6.3593e+03 |
| 17 | -1.0758e+04 | 0.0311 | -9.3288e+03 | 43 | -5.8406e+03 | 0.1672 | -6.3460e+03 |
| 18 | -1.0784e+04 | 0.0310 | -9.3211e+03 | 44 | -5.8782e+03 | 0.1661 | -6.3197e+03 |
| 19 | -1.0801e+04 | 0.0306 | -9.2948e+03 | 45 | -6.0012e+03 | 0.1639 | -6.4011e+03 |
| 20 | -1.0788e+04 | 0.0312 | -9.3298e+03 | 46 | -5.9989e+03 | 0.1634 | -6.4306e+03 |
| 21 | -1.0777e+04 | 0.0315 | -9.3132e+03 | 47 | -5.9327e+03 | 0.1645 | -6.3671e+03 |
| 22 | -1.0777e+04 | 0.0309 | -9.3824e+03 | 48 | -5.9074e+03 | 0.1659 | -6.3533e+03 |
| 23 | -1.0756e+04 | 0.0310 | -9.3781e+03 | 49 | -5.8641e+03 | 0.1675 | -6.3047e+03 |
| 24 | -1.0829e+04 | 0.0311 | -9.3636e+03 | 50 | -5.8143e+03 | 0.1703 | -6.2817e+03 |
| 25 | -1.0817e+04 | 0.0311 | -9.3187e+03 | 51 | -5.7920e+03 | 0.1700 | -6.3108e+03 |
| 26 | -1.0735e+04 | 0.0315 | -9.2620e+03 | 52 | -5.9105e+03 | 0.1656 | -6.4119e+03 |

| | 1 | 2 | 3 | | 1 | 2 | 3 |
|-----|-------------|--------|-------------|-----|-------------|--------|-------------|
| 53 | -6.0413e+03 | 0.1637 | -6.4164e+03 | 79 | -5.3981e+03 | 0.2097 | -6.1726e+03 |
| 54 | -5.8394e+03 | 0.1686 | -6.3212e+03 | 80 | -5.4273e+03 | 0.2157 | -6.1073e+03 |
| 55 | -5.7764e+03 | 0.1710 | -6.3283e+03 | 81 | -5.3903e+03 | 0.2180 | -6.0967e+03 |
| 56 | -5.9026e+03 | 0.1652 | -6.3936e+03 | 82 | -5.3733e+03 | 0.2182 | -6.0393e+03 |
| 57 | -5.3073e+03 | 0.2208 | -6.0175e+03 | 83 | -5.3930e+03 | 0.2203 | -6.0508e+03 |
| 58 | -5.2548e+03 | 0.2162 | -6.0663e+03 | 84 | -5.2885e+03 | 0.2217 | -5.9851e+03 |
| 59 | -5.3600e+03 | 0.2120 | -6.0349e+03 | 85 | -3.6644e+03 | 0.4111 | -3.9714e+03 |
| 60 | -5.4102e+03 | 0.2130 | -6.0650e+03 | 86 | -3.6275e+03 | 0.4128 | -3.9712e+03 |
| 61 | -5.3690e+03 | 0.2124 | -6.1663e+03 | 87 | -3.6289e+03 | 0.4153 | -3.9210e+03 |
| 62 | -5.3031e+03 | 0.2130 | -6.1255e+03 | 88 | -3.6836e+03 | 0.4132 | -3.9332e+03 |
| 63 | -5.4126e+03 | 0.2142 | -6.1149e+03 | 89 | -3.6362e+03 | 0.4127 | -3.9546e+03 |
| 64 | -5.3734e+03 | 0.2169 | -6.1242e+03 | 90 | -3.5456e+03 | 0.4190 | -3.8562e+03 |
| 65 | -5.3598e+03 | 0.2173 | -6.1089e+03 | 91 | -3.6259e+03 | 0.4167 | -3.9323e+03 |
| 66 | -5.3656e+03 | 0.2173 | -6.1154e+03 | 92 | -3.6428e+03 | 0.4117 | -4.0215e+03 |
| 67 | -5.3662e+03 | 0.2217 | -6.0571e+03 | 93 | -3.6391e+03 | 0.4120 | -3.9708e+03 |
| 68 | -5.3709e+03 | 0.2215 | -6.0460e+03 | 94 | -3.6128e+03 | 0.4147 | -3.9058e+03 |
| 69 | -5.3580e+03 | 0.2206 | -6.1002e+03 | 95 | -3.6408e+03 | 0.4150 | -3.9145e+03 |
| 70 | -5.3449e+03 | 0.2201 | -6.1062e+03 | 96 | -3.6794e+03 | 0.4114 | -3.9627e+03 |
| 71 | -5.3127e+03 | 0.2199 | -6.0330e+03 | 97 | -3.6614e+03 | 0.4119 | -4.0075e+03 |
| 72 | -5.3002e+03 | 0.2208 | -5.9843e+03 | 98 | -3.6397e+03 | 0.4193 | -3.9672e+03 |
| 73 | -5.2967e+03 | 0.2202 | -5.9817e+03 | 99 | -3.6719e+03 | 0.4183 | -3.9883e+03 |
| 74 | -5.3335e+03 | 0.2195 | -6.0960e+03 | 100 | -3.6166e+03 | 0.4129 | -3.9674e+03 |
| 75 | -5.3484e+03 | 0.2162 | -6.0399e+03 | 101 | -3.5866e+03 | 0.4150 | -3.9304e+03 |
| 76 | -5.4108e+03 | 0.2124 | -6.0797e+03 | 102 | -3.6076e+03 | 0.4176 | -3.9735e+03 |
| 77 | -5.3819e+03 | 0.2095 | -6.1228e+03 | 103 | -3.5671e+03 | 0.4165 | -4.0206e+03 |
| 78 | -5.3746e+03 | 0.2077 | -6.1320e+03 | 104 | -3.6430e+03 | 0.4137 | -3.9913e+03 |
| | 1 | 2 | 3 | | 1 | 2 | 3 |
| 105 | -3.6723e+03 | 0.4101 | -4.0031e+03 | | | | |
| 106 | -3.6250e+03 | 0.4166 | -3.9620e+03 | | | | |
| 107 | -3.5928e+03 | 0.4163 | -3.9822e+03 | | | | |
| 108 | -3.6426e+03 | 0.4099 | -3.9663e+03 | | | | |
| 109 | -3.6583e+03 | 0.4102 | -3.9514e+03 | | | | |
| 110 | -3.6731e+03 | 0.4090 | -3.9792e+03 | | | | |
| 111 | -3.6368e+03 | 0.4117 | -3.9510e+03 | | | | |
| 112 | -3.6032e+03 | 0.4155 | -3.9899e+03 | | | | |

Annexe (4.13)

Résultat de classification de la base de données de roulements (apprentissage et test) par le système à inférence floue FIS (optimisé paramétriquement par le système ANFIS) de la troisième étape

Remarque :

Pour Cette annexe, on a :

En ligne (une seule ligne) : sortie du système FIS (état)

En colonne (de 1 à 112) : ces colonnes portent des valeurs réelles de la sortie du système FIS (résultats de classification des observations de la base de données (apprentissage de 1 à 88 et le test de 89 à 112)).

| | | | | | | | | | | | | | | | | |
|---|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 1 | 2.0001 | 2.0001 | 2.0001 | 2.0001 | 2.0000 | 2.0002 | 2.0002 | 2.0002 | 2.0002 | 2.0002 | 2.0002 | 2.0001 | 2.0002 | 2.0002 | 2.0001 | 2.0002 |
| | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 |
| 1 | 2.0001 | 2.0002 | 2.0002 | 2.0002 | 2.0001 | 2.0002 | 3.9993 | 3.9963 | 4.0022 | 4.0006 | 4.0045 | 4.0027 | 3.9963 | 3.9982 | 3.9994 | 3.9989 |
| | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 |
| 1 | 4.0033 | 4.0006 | 3.9996 | 3.9988 | 4.0010 | 3.9969 | 3.9986 | 3.9992 | 4.0015 | 3.9979 | 3.9956 | 4.0115 | 5.9961 | 5.9995 | 6.0042 | 5.9959 |
| | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 |
| 1 | 5.9987 | 6.0029 | 6.0015 | 6.0027 | 6.0016 | 6.0020 | 6.0011 | 6.0019 | 5.9995 | 5.9968 | 5.9975 | 5.9972 | 5.9986 | 5.9962 | 6.0043 | 5.9954 |
| | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 |
| 1 | 5.9953 | 5.9934 | 7.9990 | 7.9995 | 8.0002 | 8.0004 | 7.9996 | 7.9982 | 8.0004 | 7.9987 | 7.9993 | 7.9998 | 8.0004 | 7.9993 | 7.9987 | 8.0008 |
| | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 |
| 1 | 8.0003 | 7.9995 | 7.9995 | 8.0004 | 8.0010 | 7.9995 | 7.9981 | 8.0003 | 2.0003 | 2.0002 | 2.0002 | 1.9998 | 2.0002 | 2.0002 | 4.0208 | 3.9991 |
| | 97 | 98 | 99 | 100 | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 | 111 | 112 |
| 1 | 3.9938 | 4.0033 | 4.0410 | 3.9997 | 5.9916 | 6.0042 | 6.0051 | 6.0032 | 6.0046 | 5.9939 | 8.0003 | 7.9987 | 7.9990 | 7.9982 | 7.9993 | 8.0001 |

ABSTRACT

Comparative Study of Several Optimization Methods of a Classification System of Mechanical Defects by Fuzzy Logic

The aim of this work is to optimize a diagnostic system for bearing defects based on fuzzy logic (fuzzy inference system (FIS)). For this reason, we have chosen a methodology based on three steps: the first step is to classify the different defects using an initial fuzzy inference system (FIS), whose input variables are selected randomly from a set of indicators prepared following a preliminary study of all the signals recorded from the studied system. For optimization purposes, and in the second step, an optimization is performed which affects the structure of the system and more particularly their inputs (structural optimization), selecting the best indicators in the overall set and able to separate clearly the different classes of defects studied, the selected indicators are then used as new input variables of the special diagnostic system at this second step, which is said to have optimized the structure of the diagnostic system, the method used for the automatic selection of the indicators is called the Fischer criterion method. In the third and final step, and always in order to optimize the diagnostic system (FIS), and unlike the previous step, the change this time affects the parameters of the system (parametric optimization), using a neuro-fuzzy system called ANFIS (ANFIS: Adaptive Neuro-Fuzzy Inference System), This integrated system, which combines the advantages of the two techniques, fuzzy logic (FL) and neural networks (NN), adjusts the various parameters of the functions of membership associated with the various input and output variables of the diagnostic system (FIS) by the learning process to obtain the best results in the diagnosis of defects studied.

The various programs developed during the three stages of the methodology adopted for the optimization of the system (FIS) were tested and validated using a set of vibratory signals from a test bench of bearings in different operating modes (healthy and with defects). The results obtained demonstrate the capacity and effectiveness of the methodology adopted to optimize the diagnostic system through a clear improvement of the results of the diagnosis after the optimization process compared to the results before the optimization.

Keywords: Mechanical fault diagnosis, Fuzzy logic, Fuzzy inference system (FIS), Fuzzy inference system optimization, Adaptive Neuro-Fuzzy Inference System (ANFIS), Signal processing, selection of indicators.

المخلص:

دراسة مقارنة للعديد من طرق التحسين لنظام تصنيف للعيوب الميكانيكية بواسطة المنطق الضبابي

يهدف هذا العمل إلى تحسين نظام لتشخيص عيوب المدرجات في الآلات الدوارة والذي يعتمد في بناءه على المنطق الغامض (نماذج الاستدلال الغامض (FIS)). لأجل هذا اخترنا منهجية تركز على ثلاثة مراحل، في المرحلة الأولى نقوم بإجراء تصنيف للعيوب المختلفة باستخدام نظام ابتدائي للاستدلال الغامض (FIS) تكون مداخله مختارة بطريقة عشوائية من مجموعة المؤشرات المعدة عن طريق دراسة قبلية لمجموعة الإشارات المسجلة انطلاقاً من النظام المدروس. لغرض تحسين نظام التشخيص، و خلال المرحلة الثانية، مس التحسين بنية النظام وبالضبط مداخله (تحسين بنيوي)، وذلك من خلال انتقاء أحسن المؤشرات من مجموعة المؤشرات الكلية و القادرة على فصل مختلف أصناف العيوب المدروسة فصلاً تاماً، المؤشرات المنتقاة تستعمل كمداخل جديدة لنظام التشخيص الخاص بهذه المرحلة و الذي نقول عنه أنه حسن تحسيناً بنيوياً، الطريقة المستعملة في انتقاء المؤشرات تسمى "طريقة معيار فيشر". في المرحلة الثالثة والأخيرة، ودائماً و بهدف تحسين أكبر لنظام التشخيص، و خلافاً للمرحلة السابقة مس التحسين هذه المرة معايير النظام و بالضبط معايير دوال الانتماء لمختلف مداخل و مخارج النظام و يسمى هذا التحسين بالتحسين المعياري، الطريقة المستعملة في هذا التحسين تتم باستخدام نظام مدمج يجمع المنطق الغامض مع الشبكات العصبية الاصطناعية و يجمع إيجابيات كل من التقنيتين و يسمى بنظام (ANFIS)، حيث يقوم هذا النظام المدمج و باستعمال خاصية التدريب بضبط قيم معايير دوال الانتماء لمختلف مداخل و مخارج نظام التشخيص و ذلك للحصول على أفضل النتائج أثناء التشخيص. تم الاختبار و التحقق من مختلف البرامج المعدة خلال المراحل الثلاثة لمنهجية تحسين نظام تشخيص العيوب (FIS) باستخدام مجموعة إشارات اهتزازية مسجلة انطلاقاً من بنك للتجارب خاص بالمدرجات، النتائج المتحصلة عليها أثبتت مدى قدرة و كفاءة المنهجية المعتمدة في تحسين نظام التشخيص من خلال التحسن الواضح في نتائج التشخيص بعد عملية التحسين مقارنة بالنتائج قبل التحسين.

كلمات مفتاحية: تشخيص العيوب الميكانيكية، المنطق الغامض، نظام الاستدلال الغامض، تحسين أنظمة الاستدلال الغامض، نظام الاستدلال العصبي-الغامض المكيف، معالجة الإشارة، انتقاء المؤشرات.

Résumé:

La logique floue, par sa caractéristique principale de simulation du raisonnement humain, est classée parmi les techniques de l'intelligence artificielle. Cette technique permet de modéliser, puis de remplacer l'expertise de surveillance et de conduite du processus provenant du concepteur ou de l'utilisateur. La logique floue a contribué dans le développement de plusieurs domaines. En maintenance industrielle, la logique floue est utilisée pour résoudre les problèmes du diagnostic par la classification automatique des signaux vibratoires correspondant aux différents modes de fonctionnement des machines. La technique de la logique floue(LF) souffre d'un ensemble d'inconvénients en phase de modélisation à l'aide des modèles à inférence floue à plusieurs niveaux(le choix et le nombre des variables d'entrée, le choix et le nombre des fonctions d'appartenance, la génération des règles, la méthode de défuzzification, etc). Pour surmonter ces inconvénients, on procède à une optimisation du système flou de diagnostic.

L'objectif de ce travail porte sur l'optimisation d'un système de diagnostic de défauts de roulements à base de la logique floue (système à inférence floue(SIF)). Pour cette raison, nous avons choisi une méthodologie basée sur trois étapes: la première étape consiste à classer les différents défauts à l'aide d'un système initial à inférence floue(SIF), dont les variables d'entrée sont sélectionnées de manière aléatoire à partir d'un ensemble d'indicateurs préparé suite à une étude préalable de l'ensemble des signaux enregistrés à partir du système étudié.

Pour des fins d'optimisation, et lors de la deuxième étape, on a procédé à une optimisation qui affecte la structure du système et plus particulièrement ses entrées (optimisation structurelle), en sélectionnant parmi l'ensemble des indicateurs ceux en mesure de séparer nettement les différentes classes de défauts étudiés. Les indicateurs sélectionnés sont utilisés ensuite comme nouvelles entrées du système de diagnostic spécial lors de cette deuxième étape, On dit alors que la structure du système de diagnostic a été optimisée. La méthode utilisée pour la sélection automatique des indicateurs est le critère de Fischer.

Dans la troisième et dernière étape, et toujours dans le but d'optimiser le système de diagnostic (système à inférence floue (SIF)), et contrairement à l'étape précédente, le changement affecte cette fois les paramètres du système (optimisation paramétrique), en utilisant un système neuro-flou appelé ANFIS(ANFIS : Adaptive Neuro- Fuzzy Inference System). Ce système intégré, qui combine les avantages des deux techniques , la logique

floue(LF) et les réseaux de neurones(RN), ajuste les différents paramètres des fonctions d'appartenance associées aux différentes variables d'entrée et de sortie du système de diagnostic (FIS) par le processus d'apprentissage afin d'obtenir les meilleurs résultats lors du diagnostic des défauts étudiés.

Les différents programmes développés au cours des trois étapes de la méthodologie d'optimisation du système (FIS) ont été testés et validés à l'aide d'un ensemble de signaux vibratoires d'un banc d'essais de roulements en différents modes de fonctionnement (sain et avec défauts). Les résultats obtenus démontrent la capacité et l'efficacité de la méthodologie adoptée pour optimiser le système de diagnostic grâce à une nette amélioration des résultats du diagnostic après le processus d'optimisation par rapport aux résultats avant l'optimisation.

Mots clés:

Diagnostic des défauts mécaniques, "Logique floue", Système à inférence floue(SIF), Optimisation des systèmes à inférence floue, système d'inférence neuro-flou adaptatif(ANFIS), Traitement du signal, Sélection des indicateurs.