



THÈSE



Présentée à
L'Université de Batna
Faculté de Technologie
Département de Génie Industriel
En vue de l'obtention du diplôme de

DOCTORAT EN SCIENCES

Option: Génie Industriel
par

Rafik MAHDAOUI

Magister en Génie Industriel de l'université de Batna
Ingénieur d'état en Informatique de l'université de Batna

Contribution à la surveillance dynamique des systèmes de production évolutifs par les systèmes Neuro-Flous Temporels

Thèse soutenue le :

devant le jury:

Président	N. BOUGUECHAL	<i>Professeur</i>	Université de Batna
Examineurs	M. BENMOHAMMED	<i>Professeur</i>	Université de Constantine
	M. DJOUDI	<i>Professeur</i>	Université de Poitiers
	O. KAZAR	<i>Professeur</i>	Université de Biskra
Rapporteur	L.H MOUSS	<i>Professeur</i>	Université de Batna
Co-Rapporteur	M.D MOUSS	<i>MCA</i>	Université de Batna

REMERCIEMENTS

Tout d'abord, je remercie Dieu de tout puissant de m'avoir donné le courage et la patience durant toutes ces années d'études.

Je tiens à exprimer ma profonde gratitude et ma reconnaissance envers mes promoteurs **Mme Leila Hayet MOUSS**, Professeur à l'université de Batna et directrice de Laboratoire d'Automatique et Productique « LAP » et **Mr. Mohamed Djamel MOUSS**, Maître de Conférences à l'université de Batna, de m'avoir accueilli au sein de leur équipe 3S- Surveillance – Supervision- Sûreté, pour la confiance qu'ils m'ont prodigué, pour leurs encouragements continus, leurs disponibilité, pour le suivi et la direction de mon travail, ainsi pour leurs conseils judicieux.

Par ailleurs, je remercie vivement **Mr. N. BOUGHECEHEL**, professeur à l'université de Batna, qui ma fait l'honneur de présider le jury de ma thèse.

Je tiens également à présenter mes sincères remerciements à **Mr. M. Benmouhamed**, Professeur à l'université de Constantine, **Mr. M. DJOUDI**, Professeur à l'université de Poitiers et **Mr. O. KEZZAR** professeur à l'université de Biskra, pour leur participation à l'évaluation scientifique de ce travail.

Je n'oublie évidemment pas mes collègues du LAP et mes enseignants du Département de Génie Industriel de Batna.

*A ma Mère
A mon père
A ma femme
A ma fille **Amīna** et Mon fils **Choukī Haydar**
A mes frères et mes sœurs
A la mémoire de mon frère **Choukī**
A tous ceux qui me sont chers.
Je dédie ce modeste travail*

Rafik

Résumé

Résumé

L'un des problèmes d'une grande importance dans la surveillance industrielle, est constitué par la possibilité d'effectuer une détection de fausses alarmes et éventuellement de prédire une défaillance. Ces problématiques rentrent dans le cadre de la surveillance dynamique, méthode permettant à terme de s'orienter vers le concept de maintenance proactive et, plus particulièrement, vers le pronostic. Dans ce contexte, notre travail s'oriente vers les systèmes Neuro-flous Temporels.

Ces derniers sont caractérisés par leur possibilité de représenter les connaissances d'un système de production par des connaissances symboliques et au même temps numériques dans un seul outil. Aussi, nous avons choisi d'intégrer les Systèmes d'Inférence Flous (SIF) dans une architecture de réseaux de neurones d'Elman. Le système de production choisie est l'atelier de clinkérisation de la SCIMAT, cimenterie d'Ain Touta « Batna ».

L'objectif de cette thèse consiste à faire une implémentation d'un Système d'Inférence Flou de type Takagi-Sugeno dans un Réseau de Neurones de type « Elman » pour automatiser le maximum de la surveillance dynamique d'un système de production. L'atelier de clinkérisation de la SCIMAT, cimenterie d'Ain Touta « Batna » est notre champ d'application privilégié. Afin d'atteindre cet objectif la thèse a été ainsi organisée.

La première partie présente les principales différences entre une surveillance classique et dynamique ainsi que le diagnostic prédictif, le pronostic et enfin la maintenance préventive. La deuxième partie est dédiée aux principales architectures et principes de fonctionnement des systèmes Neuro-Flous dynamique et leurs applications. Dans la troisième partie nous proposons un système Neuro-Flous dynamique pour la surveillance du système de production évolutive.

Mots clés:

Diagnostic, Réseaux de Neurones Artificiels, Logique Floue, systèmes Neuro-Flous, systèmes Neuro-Flous Temporels, Reconnaissance de Formes dynamique, Surveillance dynamique, Pronostic, Maintenance préventive.

Abstract

One issue of great importance in industrial monitoring, consists of the possibility of detection of false alarms and possibly to predict failure. These issues fall within the scope of monitoring dynamic method in terms of moving towards the concept of proactive maintenance and, more particularly, to the prognosis. In this context, our work focuses on Neuro-fuzzy systems time.

The latter are characterized by their ability to represent knowledge of a production system by symbolic knowledge and at the same time in a single digital tool. Therefore, we chose to integrate Fuzzy Inference Systems (FIS) in neural network architecture of Elman. The production system chosen IS workshop clinkering of SCIMAT, cement Ain Touta "Batna".

Our goal is to make an implementation of a fuzzy inference system of Takagi-Sugeno in a neural network type "Elman" to automate the maximum dynamic monitoring of a production system. To achieve this we organize this thesis as well.

The first part presents the main differences between conventional and dynamic monitoring and predictive diagnosis, prognosis and finally remedial maintenance. The second part is dedicated to the main architectures and operating principles of Neuro-Fuzzy dynamic systems and their applications. In the third part we propose a Neuro-Fuzzy Dynamic monitoring of the production system scalable.

Key Words:

Fault diagnosis, Artificial neural Network, Fuzzy Logic, Neuro-Fuzzy systems, pattern reconing, Temporal Neuro-Fuzzy systems, Dynamic Monitoring, Fualt Prognosis.

Table des Matières

Table des matières

Table des matières	I
Liste des figures	VI
Liste des tableaux	IX
Introduction générale	1

Chapitre I : Eléments de sûreté de fonctionnement

1.1 Introduction.....	6
1.2 Eléments de la sûreté de fonctionnement	6
1.2.1 Notions de bases et définitions.....	8
1.2.1.1 la sûreté de fonctionnement	8
1.2.1.2 la Surveillance	8
1.2.1.3 Détection	9
1.2.1.4 Diagnostic	9
1.2.1.5 Dégradation	10
1.2.1.6 Défaillance	10
1.2.1.7 Panne	10
1.2.1.8 Détection prédictive	11
1.2.1.9 Diagnostic prédictif (pronostic)	11
1.3 La maintenance industrielle	12
1.4 La surveillance industrielle : Méthodes et modèles	13
1.4.1 Méthodes de surveillance avec modèles	14

1.4.2 Méthodes de reconnaissance de formes pour la surveillance	14
1.5 Utilisation des systèmes Neuro-Flous pour la surveillance	19
1.6 Conclusion	20

Chapitre II : Le pronostic Industriel

2.1 Introduction.....	23
2.2 Le concept de pronostic	23
2.2.1 Processus de pronostic	24
2.3. Les indicateurs de pronostic	25
2.3.1 L'erreur moyenne absolue(MAPE).....	26
2.3.2 L'erreur quadratique moyenne(RMSE)	27
2.3.3 Temps restant avant défaillance –TTF-.....	27
2.3.4 L'exactitude	28
2.3.5 La Précision	29
2.3.6 Opportunité	29
2.4 Les Méthodes de pronostic industriel	30
2.4.1 Méthodes basées sur la connaissance	32
2.4.1.1 Raisonnement à base de cas	33
2.4.1.2 Les arbres de décision	33
2.4.1.3 Les systèmes flous	33
2.4.2 Méthodes guidées par les données	34
2.4.2.1 Estimation d'état et de paramètres	34
2.4.2.2 outils d'intelligence artificielle	35
2.4.3 Méthodes orientées informations	37
2.5 Pronostiquer la défaillance	38
2.5.1 Les sous-processus de pronostic	39
2.5.1.1 Piloter le pronostic	40
2.5.1.2 Initialiser l'état et les performances	41

2.5.1.3 Projeter dans le futur	41
2.5.1.4 Calculer les durées de vie résiduelle	41
2.6 Conclusion	41

Chapitre III : Les systèmes Neuro-Flous

3.1 Introduction.....	44
3.2 Les réseaux de Neuro-Flous	45
3.2.1 Définitions	45
3.3 Méthodes et Combinaisons Neuro-Floues	46
3.3.1 Le système Neuro-Flou coopératif	47
3.3.2 Le système Neuro-Flou concurrent.....	48
3.3.3Le système Neuro-Flou hybrides	48
3.3.4 Apprentissage des systèmes Neuro-Flous	53
3.4 Les avantages des systèmes Neuro-Flous	54
3.4.1La rapidité de calcul	54
3.4.2 La Flexibilité	54
3.4.3 Généralisation des connaissances	54
3.5 Les applications des systèmes Neuro-Flous	55
3.6 Conclusion	56

Chapitre IV : Les Systèmes Neuro-Flous Temporels : Outil pour la Surveillance dynamique

4.1 Introduction.....	59
4.2 Réseaux de Neurones Récurrents Elman	59
4.2.1 La structure du système Neuro-Flou Elman	60
4.2.1.1 les règles floues temporelles	60
4.2.1.2 Architectures de Réseau x Neuro-Flous Temporels	62
4.2.1.3 Algorithme d'apprentissage de SNFT	65

4.3 Test de robustesse	66
4.4 Conclusion	69
 Chapitre V : NEFSUR : une approche Neuro-Floue Proposée	
5.1 Introduction.....	71
5.2 Conception d'un outil Neuro-Flou Temporel pour la surveillance dynamique	71
5.2.1 Représentation de NEFDIAG	73
5.3 Modélisation et simulation du procédé de fabrication du Clinker	76
5.3.1 Brève présentation de l'entreprise	76
5.3.2 Simulation du fonctionnement normal	77
5.4 Pronostic de défaillance	80
5.4.1 Modèles de simulation	81
5.5 Phase de détection dynamique	87
5.5.1 Modèles de simulation de la détection	87
5.5.2 Résultats de simulation	89
5.6 Phase de diagnostic	91
5.6.1 Fonctionnement normal de l'unité de clinckersiation	93
5.6.2 Présence d'un dysfonctionnement.....	95
5.7 Conclusion	96
 Conclusion générale et perspectives	98
 Références bibliographiques	101
 Liste des publications personnelles	111
 Annexe1	115
 Annexe2	117

Listes des Figures et Tableaux

Liste des figures

Figure 1.1	Arbre de la sûreté de fonctionnement.....	07
Figure 1.2	Composantes de la surveillance industrielle.....	09
Figure 1.3	les défaillances d'après [1].....	10
Figure 1.4	Surveillance classique et surveillance dynamique.....	11
Figure 1.5	Formes de la maintenance.....	12
Figure 1.6	Classification globale des méthodes de surveillance	13
Figure 1.7	Reconnaissance des formes par RNA.....	16
Figure 2.1	Détection, diagnostic et pronostic : les aspects phénoménologiques.....	24
Figure 2.2	Loi de vieillissement pour le pronostic.....	25
Figure 2.3	TTxx et confiance d'après [14]	27
Figure 2.4	l'exactitude d'après [38].....	28
Figure 2.5	mesure de performance de système de pronostic «Timeliness ».....	30
Figure 2.6	taxonomie des approches de pronostic.....	30
Figure 2.7	la classification des méthodes de pronostic.....	32
Figure 2.8	procédure d'exécution d'un algorithme de pronostic	35
Figure 2.9	Pronostic d'un four à gas par le réseau de neurones.....	36
Figure 2.10	système intégré de maintenance prévisionnelle.....	38
Figure 2.11	environnement de processus de pronostic.....	39
Figure 2.12	décomposition du processus de pronostic.....	40
Figure 3.1	Le système Neuro-Flou.....	45
Figure 3.2	les systemes Neuro-Flous coopératifs.....	47
Figure 3.3	Les systèmes Neuro-Flous concurrents.....	48
Figure 3.4	Les Différentes architectures des systèmes neuro-flous.....	49
Figure 3.5.	Le schéma bloc du diagramme de HyFIS.....	51
Figure 3.6	L'architecture de NEFDIAG.....	53
Figure 3.6	Schéma du système d'aide au diagnostic.....	56
Figure 4.2	Architecture récurrente d'Elman.....	60
Figure 4.3	les fonctions d'appartenance utilisées.....	63
Figure 4.4	la structure de SNFT.....	64

Figure 4.5	classification des données iris.....	66
Figure 4.6	partition initiale.....	67
Figure 4.7	La base de règle générée par les données.....	68
Figure 5.1	positionnement de NEFPROG et NEFDIAG DANS NEFSUR.....	72
Figure 5.2	schéma du système d'aide à la surveillance.....	73
Figure 5.3	Schéma synoptique de l'atelier de clinkérisation.....	77
Figure 5.4	représentation de l'atelier de clinkérisation après introduction des gaz chauds et début l'arrivée de la charge.....	78
Figure 5.5	représentation de l'atelier de clinkérisation après passage de la charge dans la tour de préchauffage.....	78
Figure 5.6	Représentation de l'atelier de clinkérisation après passage de la charge dans tout l'atelier.....	79
Figure 5.7	le profil thermique la longueur du four.....	79
Figure 5.8	Détection, diagnostic and pronostic l'aspect phenomenologique	80
Figure 5.9	Le modèle de simulation.....	82
Figure 5.10	Le changement de profil de mission.....	82
Figure 5.11	série temporelle vitesse rotation four	83
Figure 5.12	LE SNFT pour la prédiction de $y(t+6)$	84
Figure 5.13	les conditions d'apprentissage et de test	84
Figure 5.14	Les fonctions d'appartenance avant apprentissage.....	85
Figure 5.15	Les fonctions d'appartenance après apprentissage.....	85
figure 5.16	la base de regle associées.....	85
Figure 5.17	comparaison de la sortie réelle avec celle donnée par le SNFT.....	86
Figure 5.18	Histogramme de l'erreur de prédiction de la vitesse rotation four.....	86
Figure 5.19	Autocorrelogramme de l'erreur de prédiction de la vitesse du four.....	86
Figure 5.20	Le Modèle de simulation pour la détection	88
Figure 5.21	les conditions d'apprentissage et de test.....	89
Figure 5.22	la base de règle Associé a une détection.....	89
Figure 5.23	Modèle normal avec 5 min de prédiction	90
Figure 5.24	Modèle normal avec 10 min de prédiction	90
Figure 5.25	Le défaut du modèle avec 10 ou 15 min de prédiction [88]	91
Figure 5.26	Effet de la panne F10 sur la rotation du four rotatif TPM [88].....	91
Figure 5.27	Exemple de message alarme (<i>decabonation perturbé</i>)	92

Figure 5.28	Exemple de message alarme (<i>mauvaise clinkerisation</i>).....	92
Figure 5.29	vitesse de rotation four en mode normale.....	95
Figure 5.30	température entrée charge et gaz du four en mode normale.....	95
Figure 5.31	vitesse de rotation four en présence d'une défaillance.....	95
Figure 5.32	température de charge et de gaz four en présence d'une défaillance.....	96

Liste des tableaux

Tableau 3.1	Comparaison entre la logique floue et les réseaux de neurones.....	46
Tableau 3.2.	Les avantages des systèmes Neuro-flous.....	55
Tableau 4.1	Les rangs de variables d'entrées « Iris ».....	67
Tableau 5.1	Description des paramètres de vecteur d'état.....	81
Tableau 5.2	modes de défaillance	87
Tableau 5.3	Description des fautes.....	90
Tableau 5.3	AMDEC de la zone surveillée	93
Tableau 5.4	Variables liées au processus de clinkérisation.....	94

Introduction générale

Introduction

Dans un grand nombre d'applications industrielles, une demande croissante est apparue en matière de remplacement des politiques de maintenance curative par des stratégies de maintenance préventive. Cette mutation d'une situation où on « subit les pannes » à une situation où on « maîtrise les pannes », nécessite quelques moyens technologiques ainsi que la connaissance de techniques d'analyse appropriées. La fonction surveillance en continu de l'évolution de l'équipement à travers des données quantifiables et qualifiables permet ainsi de prévenir un dysfonctionnement avant qu'il n'arrive et d'écartier les fausses alarmes qui peuvent ralentir la production.

Actuellement, le domaine de la maintenance a tendance à devenir un marché à part entière et beaucoup d'entreprises se spécialisant dans ce domaine ont été créées ces dernières années. La maintenance est devenue un vrai métier avec ses propres méthodologies et concepts. Parmi les facteurs qui ont favorisé cette nouvelle tendance, le développement des Sciences et Technologies de l'Information et de la Communication (STIC) prend une place très importante [3].

Par ailleurs, les composantes du système de surveillance se caractérisent par une autonomie de plus en plus importante, en travaillant dans des systèmes distribués et en intégrant souvent une intelligence embarquée. Les méthodologies de surveillance peuvent être divisées en deux grandes catégories : les méthodologies qui se basent sur l'existence d'un modèle formel de l'équipement à surveiller, les méthodologies qui se basent uniquement sur l'analyse des variables de surveillance ainsi que sur les connaissances a priori des experts humains. Les méthodes qui se basent sur une modélisation de l'équipement sont naturellement tributaires de l'existence ainsi que de la qualité d'une modélisation physique de l'équipement. Ce modèle servira de référence pour un fonctionnement nominal et tout écart par rapport au point de fonctionnement nominal sera synonyme de défaillance. L'inconvénient de ces techniques est l'existence d'incertitudes de modélisation qui sont dues au fait que la modélisation physique ne prend pas en considération tous les paramètres et les aléas qui peuvent influencer sur une information d'un paramètre de surveillance [5].

Les techniques de surveillance sans modèle sont divisées en deux parties. La première partie correspond aux outils statistiques et de traitement du signal qui sont généralement qualifiés d'outils de traitement de bas niveau, parce qu'ils sont en contact direct avec le signal capteur, et ne servent généralement que pour la génération d'alarmes brutes, sans aucune information concernant leur signification. La deuxième partie est celle des techniques dites de haut niveau et qui sont plutôt orientées vers la communication avec l'expert. Celles-ci représentent les techniques de l'Intelligence Artificielle (IA) et servent comme outil de base pour l'aide à la décision. Leur réponse est donc plus élaborée que celle des techniques de bas niveau.

Cette réponse peut être obtenue soit à partir de données brutes venant directement des variables de surveillance, soit à partir de données traitées venant de sorties des traitements de

bas niveau. Le rôle que peut jouer un expert humain reste tout de même indispensable si l'on veut concevoir un outil de surveillance avec les techniques de l'Intelligence Artificielle (IA). Parmi les techniques utilisées pour la surveillance, nos travaux concernent les Systèmes Neuro- Flous, qui se démarquent des autres outils par leur capacité d'apprentissage et de généralisation en utilisant des données numériques/ symboliques.

Les systèmes neuro-flous sont nés de l'association des réseaux de neurones et de logique floue, de manière à tirer profits des avantages de chacune de ces deux techniques.

La principale propriété des réseaux neuro-flous est leur capacité à traiter dans un même outil des connaissances numériques et symboliques d'un système [3]. Ils permettent donc d'exploiter les capacités d'apprentissage des réseaux de neurones d'une part et les capacités de raisonnement de la logique floue d'autre part.

Un système Neuro-Flou est un réseau de neurones qui est typologiquement équivalent à la structure d'un système flou. Les entrées/sorties du réseau ainsi que les poids sont des nombres réels, mais les nœuds implémentent des opérations spécifiques aux systèmes flous : fuzzyfication, opérateurs flous (conjonction, disjonction), défuzzyfication.

Objectifs du mémoire

La surveillance dynamique et surtout la prédiction d'une variable pour la maintenance prédictive nécessitent la prise en compte par le système neuro-flou de la dimension temporelle. Cette prise en compte du passé du signal n'est possible que par des architectures de systèmes neuro-flous dites temporelles. Les systèmes neuro-flous statiques sont incapables d'assurer ce genre de traitement. Une détection précoce d'un palier de dégradation ou même l'apprentissage d'une séquence d'évolution d'un système évolutif nécessite par conséquent, une architecture neuronale temporelle (dynamique). C'est dans ce contexte que se positionnent nos travaux de recherches.

La surveillance dynamique consiste à prédire une défaillance dans un Intervalle de temps suffisant pour que le superviseur « opérateur » du système puisse intervenir sans l'arrêt de ce dernier. Les problématiques posées par ce mémoire sont :

Est ce que nous pouvons faire une prévention totale suffisante dans les axes « Temps, sûreté et adaptation » pour minimiser les pannes d'un système évolutif.

Comment pouvons-nous en cas de défaillance diagnostiquer notre système ?

La reprise du système après une dégradation ou panne fonctionnelle ou matérielle.

L'objectif de notre travail est de concevoir un système de surveillance dynamique fiable permettant d'appréhender les systèmes évolutifs par les systèmes neuro-flous Temporels.

Notre recherche s'est effectuée au sein de Laboratoire d'Automatique et Productique dans le cadre de deux projets CNEPRU. Le premier intitulé "*conception d'un système de diagnostic adaptatif et prédictif pour la surveillance en ligne des systèmes évolutifs*" identifié sous le code J-0501-02-83-2006 et le second identifié sous le code B*03520100014

intitulé " *L'intelligence artificielle, les méthodes bio-inspirées, NTIC et les méthodes d'émergences : outils pour la supervision et la e-maintenance des systèmes de production*"

Notre approche consiste en l'utilisation des outils symboliques et plus précisément des méthodes de reconnaissance de formes pour l'étude d'un système dynamique d'aide à la surveillance. Ce choix est justifié par les qualités requises pour un système d'aide à la surveillance permettant ainsi l'intégration d'un apprentissage dynamique et d'une approche permettant la modélisation de l'incertitude et de l'imprécision [15].

Nous proposons une architecture de Réseaux de Neuro-Flou d'Elman qui exploite les propriétés dynamiques des architectures de réseaux de neurones d'Elman pour la prise en compte de l'aspect temporel des données d'entrée et la représentation des données symboliques de la logique floue. En effet, la prise en compte de l'aspect dynamique nécessite des architectures de réseaux de neurones particulières avec des algorithmes d'apprentissage souvent compliqués. Dans cette optique, nous proposons une version modifiée de l'algorithme de retro propagation de gradient qui permet de déterminer aisément les paramètres du réseau de Neuro-Flou.

Plan de Mémoire

Outre cette introduction, ce document comprend Cinq chapitres. Le premier chapitre traite de la problématique de la surveillance dans l'environnement industriel [15]. La surveillance industrielle est une partie intégrante de la sûreté de fonctionnement des systèmes et en constituant une fonction de plus en plus importante dans le pilotage des systèmes. Les méthodologies de surveillance sont généralement divisées en deux groupes : méthodologies de surveillance avec modèles et sans modèles. Les premières se basent sur l'existence d'un modèle formel de l'équipement et utilisent généralement les techniques de l'automatique. La deuxième catégorie de méthodologies est plus intéressante dès lors qu'un modèle de l'équipement est inexistant ou difficile à obtenir. Les besoins dans ces deux domaines évoluent constamment notamment en intelligence artificielle et exactement sur les outils Neuro-Flous pour exploiter et préserver un savoir-faire et pour amener une intelligence répartie vers les niveaux opérationnels les plus bas. De nombreux travaux ont été entrepris dans chacune de ces catégories. Nous en ferons un état de l'art de ces travaux dans ce chapitre.

Nous proposons dans le chapitre 2 d'établir un cadre applicatif au processus de pronostic en considérant les trois aspects complémentaires suivants: Comment définir le pronostic?, Comment le mesurer?, Comment le mettre en œuvre ?

D'abord, le concept de pronostic est reconsidéré : nous proposons un découpage de ce processus en deux sous-activités, la prédiction et l'évaluation. Sur cette base, différentes mesures typées pronostic sont proposées, d'abord pour caractériser le pronostic lui-même, ensuite pour juger de la qualité du processus de prédiction sous-jacent. Enfin, nous développons un guide de choix d'un outil de pronostic orienté "application" visant à permettre d'identifier une technique adaptée au contexte de l'application industrielle considérée. A cette fin, nous considérons d'une part le "cadre informationnel"

caractérisant la situation étudiée (existe-t-il ou non de la connaissance suffisante, sous quelle forme...), et d'autre part, l'applicabilité des outils en conséquence.

Le troisième chapitre est consacré à la définition des systèmes Neuro-Flous, les avantages et les inconvénients de la Logique Floue (LF) et celles des Réseaux de Neurones Artificiels (RNA), les topologies et les différentes formes de combinaisons et une présentation de la contribution des différents modèles de réseaux Neuro-Flous. Ainsi les architectures les plus connues et les plus utilisés dans l'industrie et dans les domaines de classification sont présentés.

Le cœur du travail porte sur la spécification d'un système permettant de reproduire l'évolution des propriétés d'un équipement, et de prédire un état de dégradation au cours du temps. Plus précisément les développements visent la proposition d'une architecture Neuro-Floue permettant de satisfaire un objectif de contrôle de prédiction, quel que soit l'horizon de prédiction. C'est ce qui fait l'objet du chapitre 4. A cette fin, le choix du système Neuro-Flou Temporel et exactement Système Neuro-Flou Elman (SNFE) est d'abord argumenté. Ce système Neuro-Flou Temporel est ensuite décrit et les améliorations potentielles de son aptitude à réaliser des prédictions sont identifiées. Nous proposons une solution visant à intégrer les sollicitations "futurs" connues dans le modèle prédictif.

Le dernier chapitre est dédié à la présentation des résultats sur laquelle a débouché notre étude. En effet, nous proposons une stratégie pour le suivi du comportement d'un processus et la détection des défaillances. Une approche de surveillance industrielle basée sur la reconnaissance de formes Neuro-Floue qui s'appuie sur une représentation numérique et au même temps symbolique des formes est alors mise au point.

Afin d'implanter cette approche et l'exploiter, pour surveiller un système de production, nous avons développé un logiciel informatique que nous avons nommé SURNEF (SURveillance industrielle par les systèmes Neuro-Flous Temporels)

Le SURNEF est un logiciel informatique de simulation interactive réalisé au sein de LAP, pour le développement, l'apprentissage et le test d'un système Neuro-Flou Temporel de prédiction des fautes, de détection dynamique et de classification des pannes d'un procédé industriel. SURNEF peut être représentée comme un type spécial de perceptron flou récurrent, à Cinq couches utilisées pour la prédiction des pannes et ensuite pour classifier des formes et des défaillances.

Enfin la dernière partie conclue ce mémoire. Nous y dresserons un bilan final de notre travail et nous y donnerons quelques perspectives de recherche prometteuses.

Chapitre I

Eléments de sureté de fonctionnement

Chapitre I

Eléments de sûreté de fonctionnement

1.1 Introduction

La sûreté de fonctionnement consiste à connaître, évaluer, prévoir, mesurer et maîtriser les défaillances des systèmes technologiques et les défaillances humaines. Elle est appelée "science des défaillances". D'autres désignations existent suivant les domaines d'applications : analyse de risque (milieu pétrolier), cyndinique (science du danger), FMDS (Fiabilité, Maintenabilité, Disponibilité, Sécurité), en anglais RAMS (Reliability, Availability, Maintainability and Safety).

Elle se caractérise à la fois par les études structurelles statiques et dynamiques des systèmes, du point de vue prévisionnel mais aussi opérationnel et expérimental (essais, accidents), en tenant compte des aspects probabilistes et des conséquences induites par les défaillances techniques et humaines. Cette discipline intervient non seulement au niveau de systèmes déjà construits mais aussi au niveau conceptuel pour la réalisation de systèmes [15].

Afin de mieux percevoir la problématique traitée, nous avons jugé important d'établir un lexique des termes utiles dans cette thèse. Les définitions ci-après sont extraites de normes françaises et internationales, ainsi que de l'étude d'ouvrages synthétiques.

1.2 Éléments de la sûreté de fonctionnement

Le terme Sûreté de Fonctionnement (SdF) est multi facette. Il recouvre trois classes de notions : les entraves, les mesures et les moyens. La figure 1.1 résume les différents aspects exposés de la sûreté de fonctionnement.

Les entraves à la SdF regroupent tous les éléments venant altérer la production. *Les mesures*, ou les attributs, de la SdF expriment à la fois les propriétés attendus du système de production et leur appréciation effective et en fin, *les moyens* qui constituent les méthodes et les techniques pour conférer au système de production l'aptitude à délivrer le service spécifié, Et d'autre part pour donner confiance en cette aptitude, c'est-à-dire la valider.

Les moyens d'obtention de la SdF peuvent se résumer dans l'évitement de la création d'entraves et à la correction d'entraves lorsque ces derniers n'ont pas pu être évitées. Aussi, deux types de traitement sont alors préconisés :

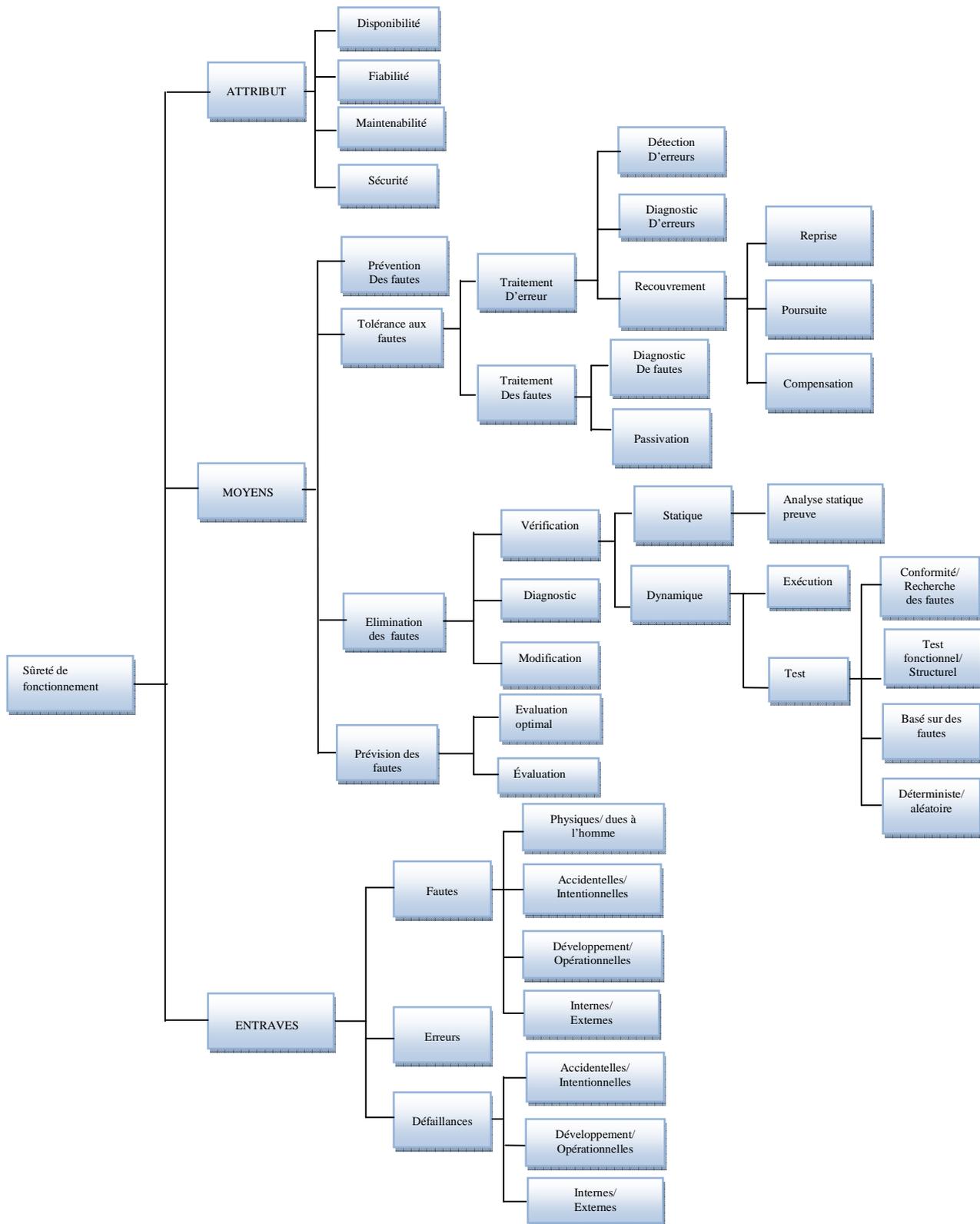


Figure 1.1 Arbre de la sûreté de fonctionnement

- La prévention de fautes : ce type d'approche consiste notamment à développer des outils de modélisation capables de considérer plus d'informations, avec des règles d'utilisation plus strictes, ce qui limite les fautes d'utilisation, malheureusement, il n'est pas possible d'éliminer toutes les fautes puisque le cycle de développement de ces outils et lui-même entaché de fautes. Ce qui signifie que les outils développés ne seront peut-être pas totalement adaptés au problème auquel ils ont été appliqués. Des défaillances peuvent alors se produire, qu'il faut traiter avec des méthodes correctives.
- La tolérance aux défaillances : elle assure que le système fournit le produit prévu en dépit de défaillances. Il s'agit de traitements appliqués aux défaillances. Ils doivent corriger la défaillance, soit en réparant l'élément défaillant, soit en utilisant des redondances (de matériel, de logiciel...). Dans la mesure du possible, la faute doit aussi être corrigée de manière à ce qu'elle ne provoque plus de défaillance.

A ces traitements d'obtention de SdF d'un système, viennent s'ajouter les moyens pour déterminer le degré de la SdF du système de production ainsi conçu, et donc indiquer s'il est possible d'avoir dans l'aptitude du système à délivrer un service conforme au service spécifié. Ces traitements n'agissant pas sur le système, ils ne participent pas directement à l'amélioration de la SdF. Ils y contribuent toutefois lorsque les résultats qu'ils fournissent sont utilisés pour modifier le système avant son utilisation définitive [16].

1.2.1 Notions de bases et définitions

1.2.1.1 la sûreté de fonctionnement

La sûreté de fonctionnement est l'aptitude d'une entité à satisfaire une ou plusieurs fonctions requises dans des conditions données. On notera que ce concept peut englober la fiabilité, la disponibilité, la maintenabilité, la sécurité, la durabilité... ou des combinaisons de ces aptitudes. Au sens large, la Sûreté de fonctionnement est considérée comme la science des défaillances et des pannes [14]. La sûreté de fonctionnement est souvent définie comme :

- FMDS (Fiabilité, Maintenabilité, Disponibilité et Sécurité);
- Science des défaillances;
- Analyse de risque.

1.2.1.2 la Surveillance

La surveillance est un dispositif passif, informationnel, qui analyse l'état du système et fournit des indicateurs. La surveillance consiste notamment à détecter et classer les défaillances en observant l'évolution du système, puis à les diagnostiquer en localisant les éléments défaillants et en identifiant les causes premières.

La surveillance se compose donc de deux fonctions principales qui sont la *détection* et le *diagnostic* (voir la figure 1.2):

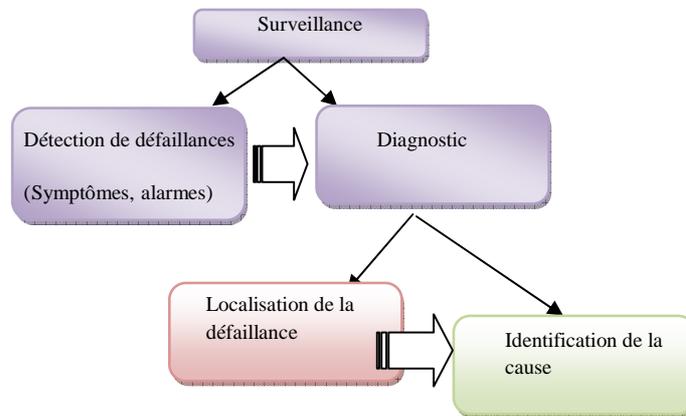


Figure 1.2 Composantes de la surveillance industrielle

1.2.1.3 Détection

Pour détecter les défaillances du système, il faut être capable de classer les situations observables comme étant normales ou anormales.

1.2.1.4 Diagnostic

L'objectif de la fonction diagnostic est de rechercher les causes et de localiser les organes qui ont entraîné une observation particulière. La fonction de diagnostic se décompose en deux fonctions élémentaires : localisation et identification des causes.

La localisation permet de déterminer le sous-ensemble fonctionnel défaillant tandis que l'identification de la cause consiste à déterminer les causes qui ont mené à une situation anormale.

A partir de l'observation d'un état de panne, la fonction diagnostic est chargée de retrouver la faute qui en est à l'origine. Ce problème est difficile à résoudre. En effet si, pour une faute donnée, il est facile de prédire la panne résultante, la démarche inverse qui consiste à identifier la faute à partir de ses effets, est beaucoup plus ardue. Une défaillance peut généralement être expliquée par plusieurs fautes. Il s'agit alors de confronter les observations pour fournir la bonne explication.

1.2.1.5 Dégradation

Une dégradation représente une perte de performances d'une des fonctions assurées par un équipement.

Si les performances sont au-dessous du seuil d'arrêt défini dans les spécifications fonctionnelles de cet équipement, il n'y a plus dégradation mais défaillance.

1.2.1.6 Défaillance

Une défaillance est l'altération ou la cessation de l'aptitude d'un ensemble à accomplir sa ou ses fonction(s) requise(s) avec les performances définies dans les spécifications techniques. L'ensemble est indisponible suite à la défaillance.

Un système peut remplir sa fonction tout en présentant une anomalie de comportement. Par exemple, une machine électrotechnique peut produire un bruit anormal tout en entraînant correctement une charge, en supposant que telle soit sa fonction. Le bruit anormal est un défaut qui peut permettre de présager d'une défaillance à venir figure 1.3 [14].

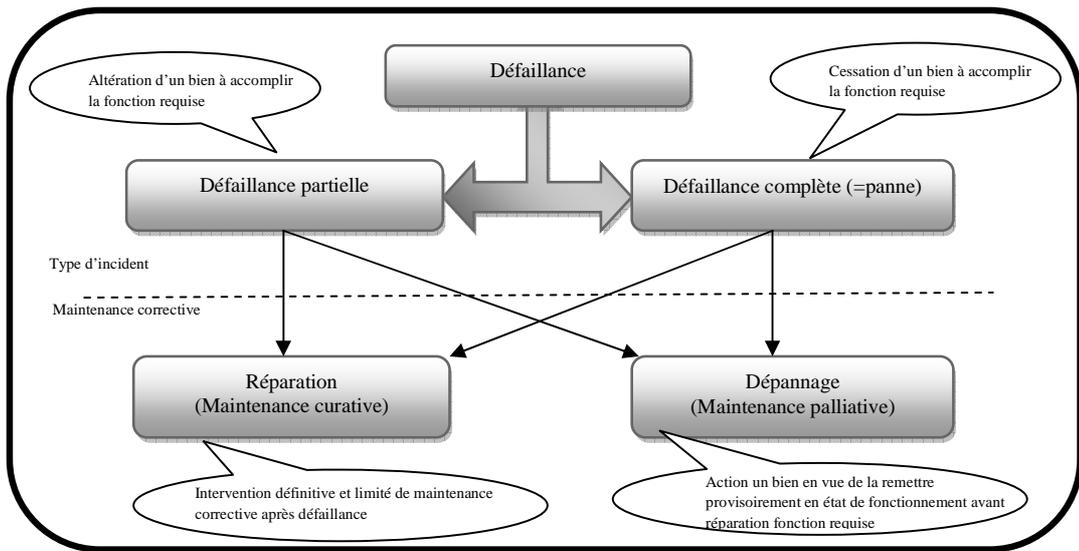


Figure 1.3 Les défaillances d'après [1]

1.2.1.7 Panne

Une panne est l'inaptitude d'une entité (composant ou système) à assurer une fonction requise. A partir du lexique des différents termes donnés, on peut constater qu'il y a une différence entre la surveillance classique et la surveillance dynamique. Cette différence se trouve dans la figure 1.4.

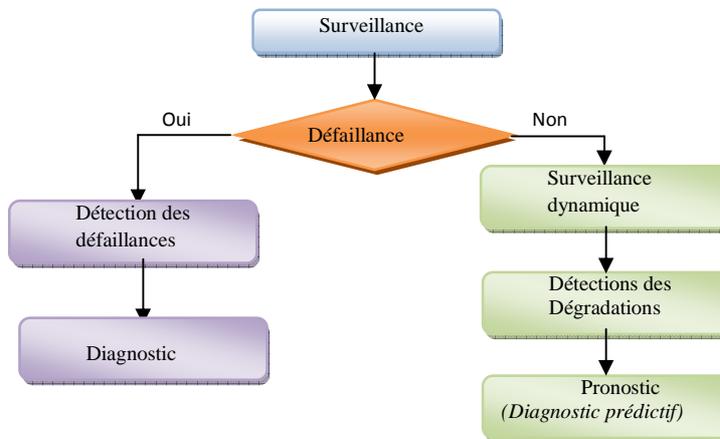


Figure 1.4 Surveillance classique et surveillance dynamique

Comme pour la surveillance classique, La surveillance prédictive est un dispositif passif, informationnel, qui analyse l'état présent et passé du système et fournit des indicateurs sur les tendances d'évolution future du système. La surveillance prédictive se compose de : la détection prédictive (dynamique) et du diagnostic prédictif, appelé aussi pronostic.

1.2.1.8 Détection prédictive

La détection prédictive consiste à prédire une défaillance future. En d'autres termes, le but de la détection prédictive est de détecter une dégradation au lieu d'une défaillance.

1.2.1.9 Diagnostic prédictif (pronostic)

L'objectif du diagnostic prédictif est d'identifier les causes et de localiser les organes qui ont entraîné une dégradation particulière.

Si les performances sont au-dessous du seuil d'arrêt défini dans les spécifications fonctionnelles de l'équipement, il n'y a plus dégradation mais défaillance.

Donc, la Sdf vise notamment à mettre en évidence dès la phase de conception, la meilleure fiabilité et disponibilité qui conduisent à des systèmes de production à la fois non dangereux et rentables en quantité et en qualité.

Le système qui a un fonctionnement sûr est ainsi un système qui réalise ce pour quoi il a été conçu, sans incident mettant sa rentabilité en question et sans incident mettant la sécurité en jeu.

1.3 La maintenance industrielle

La maintenance est définie comme l'ensemble de toutes les actions techniques, administratives et de management durant le cycle de vie d'un bien, destiné à le maintenir ou à le rétablir dans un état dans lequel il peut accomplir la fonction requise.

La maintenance industrielle qui comprend ainsi un ensemble d'actions de dépannage, de réparation, de contrôle et de vérification des équipements matériels, et doit contribuer à l'amélioration des processus industriels. Dans la vision traditionnelle, la fonction maintenance permet de garantir les caractéristiques de sûreté de fonctionnement des matériels, notamment la disponibilité. Les principaux termes permettant d'en apprécier l'objectif sont les suivants :

Sûreté de fonctionnement : ensemble des propriétés qui décrivent la disponibilité et les facteurs qui la conditionnent : fiabilité, maintenabilité, et logistique de maintenance.

Disponibilité : aptitude d'un bien à être en état d'accomplir une fonction requise dans des conditions données, à un instant donné ou durant un intervalle de temps donné, en supposant que la fourniture des moyens extérieurs nécessaires est assurée.

Bien : tout élément, composant, mécanisme, sous-système, unité fonctionnelle, équipement ou système qui peut être considéré individuellement.

Fiabilité : aptitude d'un bien à accomplir une fonction requise, dans des conditions données, durant un intervalle de temps donné.

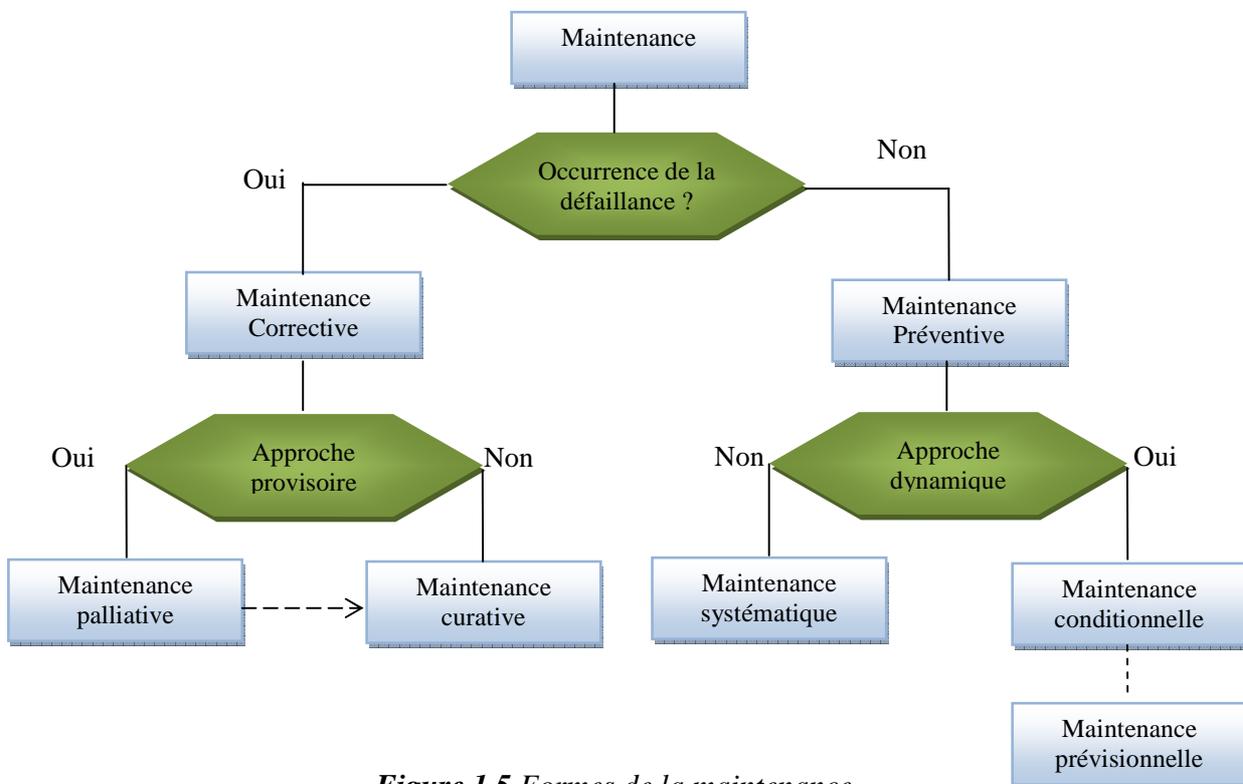


Figure 1.5 Formes de la maintenance

Ces deux grands types de maintenance, corrective et préventive, présentent certaines variantes exposées ci-après. La figure 1.5 en donne une articulation globale [03].

1.4 La surveillance industrielle : Méthodes et modèles

Les méthodes de surveillance industrielle telle qu’elles sont présentées dans ce paragraphe sont illustrées sur la Figure 1.6. L’existence d’un modèle formel ou mathématique de l’équipement détermine la méthode de surveillance utilisée. La surveillance avec modèle se compose essentiellement de deux techniques : méthodes de redondance physique et analytique et méthodes d’estimation paramétrique.

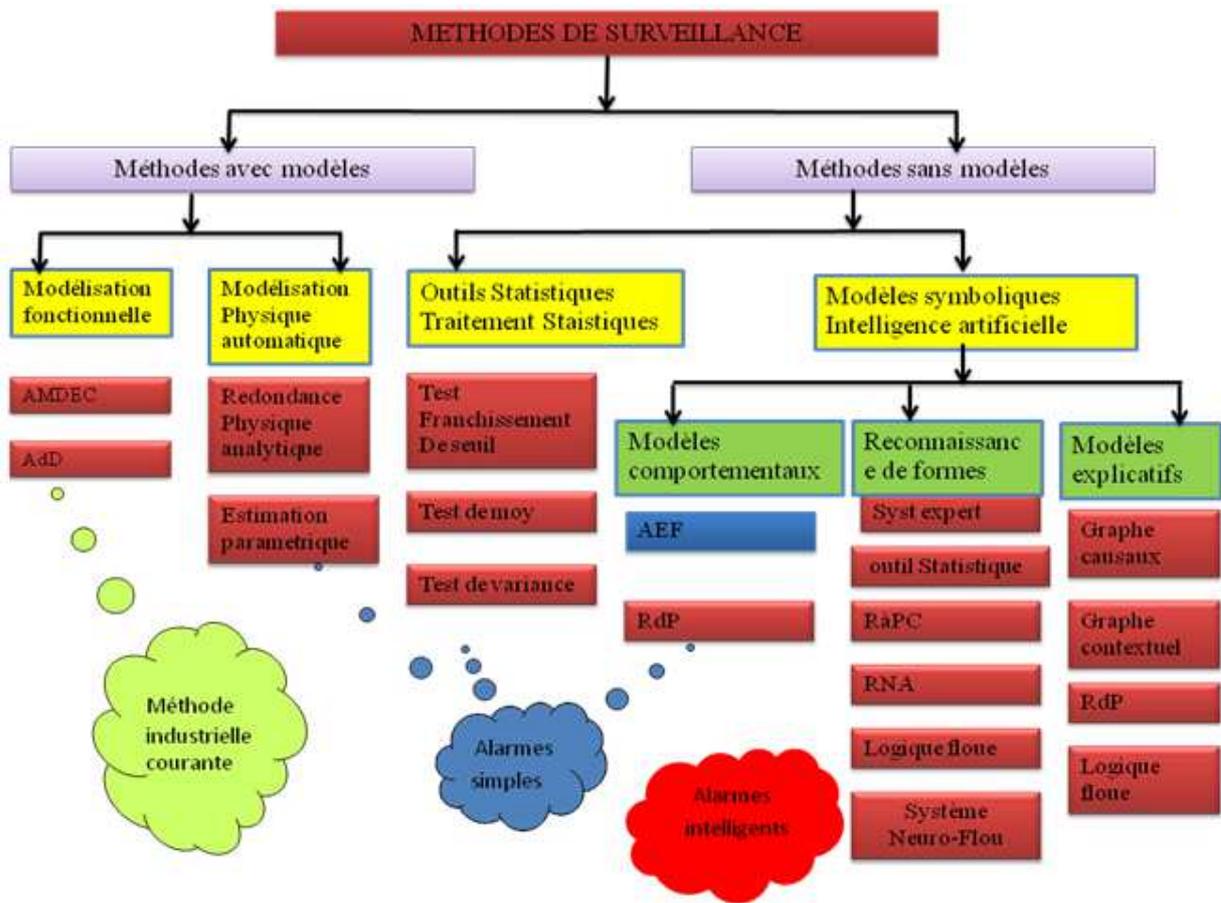


Figure 1.6 Classification globale des méthodes de surveillance

D’un autre côté, les méthodes qui ne se basent pas sur l’existence du modèle se divisent en deux catégories : méthodes utilisant des outils statistiques et méthodes de reconnaissance des formes. Les outils statistiques établissent des tests sur les signaux d’acquisition. Ces tests ne sont capables d’assurer que la fonction détection de défaillances. Par contre, les

techniques de surveillance par reconnaissance des formes sont plus élaborées par rapport aux simples tests statistiques et sont capables de détecter et de diagnostiquer les défaillances.

1.4.1 Méthodes de surveillance avec modèles

La catégorie de méthodes les plus utilisées par les industriels à l'heure actuelle est vraisemblablement celle des méthodes par modélisation fonctionnelle et matérielle. Cette catégorie de méthodes comprend des outils industriels comme l'AMDEC (Analyse des Modes de Défaillance, de leurs Effets et de leur Criticité) et AdD (l'Arbre de Défaillances) utilisés essentiellement pour le diagnostic.

Les méthodes de diagnostic par modélisation fonctionnelle et matérielle

Le principe de ces méthodes consiste à établir a priori et de la manière la plus complète possible, les liens entre les causes initiales des défaillances et leurs effets mesurables.

Une autre catégorie de méthodes avec modèles est représentée par les méthodes basées sur une modélisation physique des processus surveillés.

Les méthodes de surveillance par modélisation physique

Les méthodes de surveillance avec modèle physique ont pour principe de comparer les mesures effectuées sur le système aux informations fournies par le modèle (Frank, 1990). Tout écart est alors synonyme d'une défaillance. Les outils de la théorie de la décision sont ensuite utilisés pour déterminer si cet écart est dû à des aléas normaux, comme par exemple le bruit de mesure ou s'il traduit une défaillance du système. Les méthodes avec modèle physique sont les méthodes les plus familières aux automaticiens. En règle générale, ces méthodes peuvent être séparées en deux techniques : techniques de redondance physique et analytique, et techniques d'estimation paramétrique [3].

1.4.2 Méthodes de reconnaissance de formes pour la surveillance

Ces méthodes supposent qu'aucun modèle n'est disponible pour décrire les relations de cause à effet. La seule connaissance repose sur l'expertise humaine confortée par un solide retour d'expérience [18]. La plupart de ces méthodes sont basées sur IA. Ces méthodes regroupent les modèles associatifs et les méthodes de reconnaissance dans le sens où elles sont caractérisées par les termes apprentissage et reconnaissance qui s'appliquent aussi bien aux systèmes de reconnaissance de formes qu'aux systèmes à base de règles tels que les systèmes experts. Dans ces méthodes on considère particulièrement les outils suivants :

- les réseaux neuronaux,
- la logique floue,
- les réseaux Neuro-flous,
- les systèmes experts,

- le raisonnement à partir de cas,
- les outils statistiques.

a. Les réseaux de neurones

Les réseaux de neurones sont des outils capables d'effectuer des opérations de perception, classification et prédiction. Leur fonctionnement est basé sur le principe de fonctionnement des neurones biologiques. Leur principal avantage par rapport aux autres outils est leur capacité d'apprentissage et de généralisation de leurs connaissances à des entrées inconnues.

Une des qualités de ce type d'outil est son adéquation pour la mise au point de systèmes de surveillance modernes, capables de s'adapter à un système complexe avec reconfigurations multiples. Les réseaux de neurones peuvent être également implémentés en circuits électroniques, offrant ainsi la possibilité d'un traitement temps réel.

Leur utilisation est principalement guidée par leurs propriétés suivantes:

- capacité d'apprentissage,
- capacité de généralisation,
- parallélisme dans le traitement (rapidité de traitement)
- adaptés aux non-linéarités des systèmes,

Chaque neurone réalise une fonction simple (fonction linéaire, linéaire par morceaux, fonction à seuil, sigmoïde, gaussienne), les propriétés globales de l'outil émergent de sa structure. Toutes les caractéristiques des réseaux de neurones sont exploitées à travers la propriété principale des réseaux de neurone qu'est l'apprentissage. En effet, les mécanismes d'apprentissage sont à l'origine des capacités de résolution de problèmes des réseaux neuronaux. Cet apprentissage permet de configurer les poids synaptiques ainsi que les fonctions d'activation afin d'adopter un comportement désiré. Deux types d'apprentissages sont utilisés : l'apprentissage supervisé et l'apprentissage non supervisé [16] [17].

Les architectures les plus utilisées pour ces problèmes de surveillance des systèmes industriels sont le modèle de Hopfield (basé sur la mémoire associative), le réseau de Kohonen (carte topologique auto-adaptative), le Perceptron Multicouches (PMC ou MLP pour Multi layer Perceptron) et le Réseau à Fonctions de base Radiales (RFR ou RBF pour Radial Basis Function).

La figure 1.7 montre l'architecture générale d'une application de surveillance par reconnaissance de formes avec réseaux de neurones. L'expert humain joue un rôle très important dans ce type d'application. Toute la phase d'apprentissage supervisé du réseau de neurones dépend de son analyse des modes de fonctionnement du système. Chaque mode est caractérisé par un ensemble de données recueillies sur le système.

A chaque mode on associe une expertise faite par l'expert. Cette association (ensemble de données -modes de fonctionnement) sera apprise par le réseau de neurones. Après cette phase d'apprentissage, le réseau de neurones passe en phase de classification, dans laquelle il associera les classes représentant les modes de fonctionnement aux formes d'entrée caractérisées par les données du système.

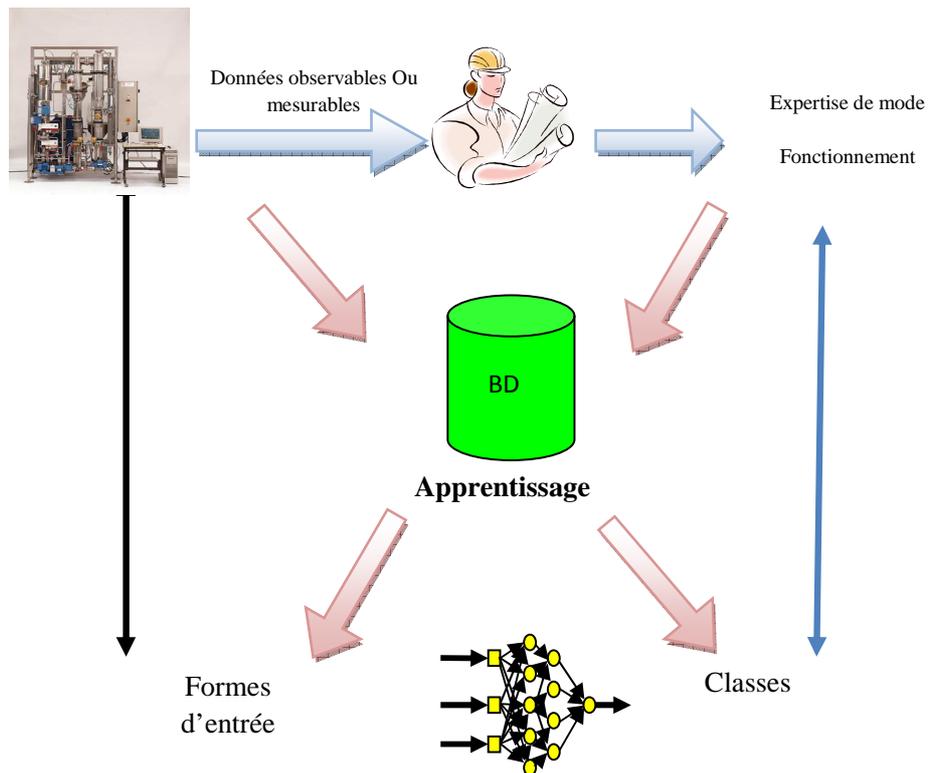


Figure 1.7 Reconnaissance des formes par RNA

Parmi les réseaux de neurones cités, les réseaux de Kohonen connaissent de nombreuses applications pour la classification de signatures vibratoires de machines. Ils présentent cependant l'inconvénient majeur d'exiger un temps relativement long pendant la phase d'apprentissage [15].

Le modèle perceptron multicouche se caractérise donc par une architecture globale (tous les neurones participent à la sortie après la phase d'apprentissage), qui lui confère de mauvaises propriétés en classification et donc le rend moins intéressant pour une utilisation en surveillance. Cependant il existe un nombre non négligeable de travaux en surveillance et diagnostic qui utilisent cette architecture, de nombreuses références sont fournies dans [25].

Celui qui semble se prêter le mieux à l'utilisation en surveillance est le réseau à base de fonctions radiales (RFR – en anglais Radial Basis Function - RBF). Ceci est du en grande partie

à son approche locale et à sa capacité de travailler dans un « monde ouvert », en enrichissant son expérience au fur et à mesure de son fonctionnement.

Les applications précédentes montrent que les réseaux de neurones peuvent fournir des solutions très intéressantes pour les problèmes de surveillance sans modèle. Toutefois, un aspect fort important n'a pas encore été abordé jusque là, à savoir la dynamique du système à surveiller. En effet, toutes les applications présentées ne traitent que l'aspect statique des données capteurs. Ce type de traitement ne permet pas d'apprendre la dynamique d'un signal capteur afin de prédire une défaillance, ni même d'apprendre des séquences de fonctionnement d'un système à événements discrets. Les réseaux de neurones temporels offrent cette possibilité de prise en compte de l'aspect temporel des données.

b. La logique floue

La logique floue permet de formaliser la représentation et le traitement de connaissances imprécises ou approximatives. Elle offre la possibilité de traiter des systèmes d'une grande complexité dans lesquels sont, par exemple, présents des facteurs humains. Elle intervient dans la manipulation de connaissances imparfaites. Son utilisation dans des domaines tels que l'aide à la décision ou le diagnostic semble donc naturelle dans la mesure où elle fournit un outil puissant pour assister de façon automatique des actions humaines, naturellement empreintes d'imprécisions. La logique floue est ainsi considérée (Bouchon-Meunier,1994,1995) comme le seul cadre dans lequel on peut traiter des imprévus et des incertitudes, qui autorise également le traitement de certaines incomplétudes, et le seul cadre dans lequel on peut traiter des connaissances numériques et des connaissances exprimées symboliquement par des qualifications du langage naturel.

Les applications de la logique floue sont extrêmement nombreuses et variées. Les plus courantes sont la commande floue, les systèmes experts flous, le raisonnement à partir de cas et la reconnaissance floue de formes. Dans le cadre de la surveillance et du diagnostic, on trouve principalement les systèmes experts, le raisonnement à partir de cas et la reconnaissance de formes. Dans ces différents contextes (aide au diagnostic, aide à la décision), l'expert humain exprime des connaissances ou des données dans un langage naturel fondamentalement imprécis; la logique floue permet donc d'une part de prendre en compte les imprécisions inhérentes aux données et d'autre part de rendre compte de l'expression des règles qui permettent de formuler un diagnostic ou de déterminer une action. On trouve par exemple dans [27] l'architecture d'un outil de détection/diagnostic d'antennes, dans lequel la logique floue intervient sous forme d'un système expert flou et dans les étapes de classification :

Dans le cadre de la surveillance des systèmes industriels, la logique floue se trouve également associée à d'autres outils et techniques d'analyses. A titre d'exemple, la logique floue

a trouvé des applications en combinaison avec les arbres de défaillances afin d'évaluer l'apparition de l'événement sommet [24].

Dans ces diverses applications, l'utilisation de la logique floue est assez naturelle dans la mesure où elle permet de traiter l'imprécision, l'incertitude et l'incomplétude liées aux connaissances du domaine. En plus, la logique floue leur confère une capacité d'utilisation en pronostic. Cependant, même si la logique floue fournit des résultats satisfaisants, on ne peut considérer ces applications comme de véritables applications de la logique floue pour le diagnostic dans la mesure où ces différents outils ne s'appliquent pas à la localisation et à l'identification des causes expliquant un défaut. Utilisée avec les arbres de défaillances, cette dernière devrait fournir une évaluation sur l'occurrence ou la présence des événements de base de l'arbre de défaillances qui sont eux à l'origine de l'événement sommet. On obtiendrait ainsi l'évaluation des causes à l'origine d'un dysfonctionnement.

c. Les systèmes Neuro-Flous

Les principaux avantages des techniques floues sont l'approche naturelle de la modélisation et la bonne interopérabilité de la description, en employant des règles linguistiques. Cependant, comme il n'y a aucune méthode formelle pour déterminer ses paramètres (ensembles et règles floues), l'exécution d'un système flou peut prendre beaucoup de temps. Dans ce sens, il serait intéressant de disposer d'algorithmes permettant l'apprentissage automatique de ces paramètres.

L'une des méthodes qui permet de répondre à ces exigences est la théorie des réseaux de neurones qui emploie des échantillons (données d'observation) pour l'apprentissage. La combinaison des deux techniques nous donne les systèmes Neuro-Flou.

Différentes combinaisons de ces deux techniques d'IA existent et mettent en avant des propriétés différentes, on peut identifier les combinaisons suivantes :

- Réseau flou neuronal
- Système neuronal/flou simultanément
- Modèles Neuro-Flous coopératifs
- Modèles Neuro-Flous hybrides

Les réseaux Neuro-Flous apparaissent comme des outils puissants combinant des grandes capacités d'approximation pour la modélisation des systèmes dynamiques non-linéaires pour lesquels le modèle mathématique est inconnu avec la possibilité d'obtenir des résultats possédant un certain niveau d'interprétation.

La plupart des applications rencontrées sont basées sur l'établissement d'un diagnostic à partir de la classification de résidus, elles nécessitent donc de pouvoir établir un modèle du système. De plus, elles nécessitent un réseau par résidus ce qui rend le système relativement complexe et ne permet de diagnostiquer qu'un nombre restreint de défauts. Il serait

donc intéressant d'employer ces techniques compte tenu de leurs capacités en s'affranchissant complètement d'un modèle du système à diagnostiquer.

1.5 Utilisation des systèmes Neuro-Flous pour la surveillance

Dans la bibliographie liée à l'utilisation des systèmes Neuro-Flous pour le diagnostic, nous avons recensé quelques contributions importantes :

- Dans [51], les auteurs présentent une méthode utilisant la logique floue pour le diagnostic des transformateurs. Ils envisagent une utilisation ultérieure des réseaux de neurones pour la génération automatique de règle à partir des données.
- Dans [28], l'auteur associe un modèle dynamique des liens causaux entre les défaillances, au modèle numérique de traitement du signal. Ce choix permet la prise en compte d'informations liées à la fiabilité, en plus des informations en provenance des capteurs. Les outils utilisés sont les réseaux de Pétri et les chaînes de Markov Cachées (HMM – Hidden Markov Model).
- Dans [29], l'auteur présente une méthode de diagnostic réparti pour les systèmes à événements discrets comme les réseaux de télécommunication. Une technique de fusion des diagnostics locaux est utilisée pour obtenir un diagnostic global.
- Dans [15], l'auteur propose l'utilisation de techniques Neuro-floues temporels pour une aide dynamique au diagnostic. Cette méthode comprend plusieurs étapes : L'acquisition des informations pertinentes du système, Application du système de détection basé sur les réseaux de neurones dynamiques, Application du système de diagnostic basé sur un réseau Neuro-flou.
- Dans [14], L'auteur propose une proposer un Neuro-Flou temporel permettant de prédire la dégradation d'un équipement sans connaissance a priori sur son comportement, et de générer les indicateurs de pronostic permettant d'optimiser les stratégies de maintenance.

Pour développer les systèmes de surveillance dynamique par les systèmes Neuro-Flous Temporels [15]. Il existe deux façons distinctes d'aborder le temps par les réseaux de Neuro-Flous appliquées a la surveillance industrielle ; dans la première, le temps est représenté comme un mécanisme externe au RNF, Cette technique est aussi appelée représentation spatiale du temps. Par contre, dans la deuxième façon de prendre en compte le temps, le RNF est capable de traiter le temps sans aucun mécanisme externe Cette représentation est appelée représentation interne. On peut également voir que la représentation interne se divise en deux possibilités : soit que le temps est pris en compte implicitement par la récurrence des connexions (réseaux de neurones récurrents), soit qu'il est pris en compte d'une manière explicite.

1.6 Conclusion

Dans ce chapitre, nous avons pu abordé le point de vue de la sûreté de fonctionnement où nous avons présenté divers outils nécessaires pour améliorer ses quatre caractéristiques, la fiabilité, la maintenabilité, la disponibilité et la sécurité.

Les méthodes de la surveillance présentées dans ce chapitre ont pour finalité d'aider les experts « exploitants » à prendre rapidement des décisions sûres pour optimiser la disponibilité des moyens de production.

L'utilisation des techniques de surveillance avec modèles semble de plus en plus difficile et coûteuse, en offrant des performances de moins en moins satisfaisantes. En plus, la surveillance à base de modèle du système est souvent opérée hors ligne, empêchant ainsi des traitements en temps réel de l'information, une exigence de plus en plus présente chez les industriels.

Nous en avons conclu que les techniques issues de l'IA constituent la meilleure solution pour l'aide à la décision pour l'opérateur de maintenance. Donc, nous avons besoin d'outils qui devront exploiter aussi bien le savoir-faire des automaticiens que les données des méthodes de SdF, dans un cadre de surveillance industrielle, l'IA offre des outils totalement découplés de la structure du système, permettant un suivi temps réel de l'évolution de celui-ci. Le raisonnement en ligne fait que l'approche de l'IA est plus robuste à des changements de modes opératoires comme pour les systèmes ayant plusieurs configurations. Elle est donc évolutive.

Les techniques de détection dynamique de dégradations/défaillances font appel à des méthodes de reconnaissances de formes pour le traitement dynamique des informations. En ce qui concerne l'aide au diagnostic et au pronostic, le rôle central est joué par la recherche de la cause de la défaillance ou de la dégradation. Ce volet de la surveillance ne peut pas être traité par les techniques de reconnaissances de formes et nécessite l'utilisation de modèles explicatifs (données numériques ou/et symboliques).

Dans ce sens, nous nous sommes intéressés à la modélisation de la propagation des défaillances et à la démarche de recherche de cause à l'aide des systèmes Neuro-Flous.

Ces derniers sont capables grâce à la logique floue de prendre en compte aussi bien l'expérience humaine et l'incertain sous la forme de règles de propagation de défauts ou règles de recherche de cause, ainsi que de prendre en compte grâce à sa composante neuronale la notion d'apprentissage et l'adaptation à un contexte de surveillance évolutif.

Ainsi, la thématique de pronostic devient un cadre de travaux de premier plan et les avantages potentiels de l'implémentation du pronostic dans des milieux industriels, liés à la sécurité du travail, aux aspects économique et aux ressources humaines ont poussé les scientifiques à s'y intéresser. Aujourd'hui le "pronostic" est considéré comme un processus clef dans les stratégies de maintenance.

De nombreux outils et méthodes de pronostic de défaillance ont été proposés durant la dernière décennie. Les méthodes de pronostic diffèrent généralement par le type d'application considérée, alors que les outils mis en œuvre dépendent principalement de la nature des données et connaissances disponibles pour construire un modèle de comportement du système réel incluant le phénomène de la dégradation. Aussi, ces méthodes et outils peuvent être regroupés dans un nombre limité d'approches.

Nous nous sommes intéressés plus particulièrement dans le chapitre suivant à la description de différentes méthodes de formalisation et de modélisation du pronostic.

Chapitre II

Le Pronostic Industriel

Chapitre II

Le pronostic Industriel

2.1 Introduction

L'activité de pronostic industriel est considérée comme un processus clef dans les stratégies de maintenance industrielle. Une évolution particulière porte sur la manière d'appréhender les phénomènes de défaillance : peu à peu les industriels tendent, non seulement à les anticiper afin de recourir à des actions préventives, mais en plus à le faire de manière la plus juste possible dans un objectif de réduction des coûts et des risques. Cette évolution a donné une part grandissante au processus de pronostic qui est aujourd'hui considéré comme un des principaux leviers d'action dans la recherche d'une performance globale.

De manière plus ciblée, le concept de pronostic ainsi que les outils et méthodes de pronostic sont également arborés. L'ensemble permet de montrer la diversité des approches envisageables et de définir un cadre de travail plus restreint pour cette thèse.

Ce chapitre vise à présenter l'émergence du pronostic, sa définition et son positionnement vis-à-vis des stratégies de maintenance. La deuxième partie est consacrée à la description des mesures utilisées pour la prédiction et l'évaluation du processus de pronostic.

2.2 Le concept de pronostic

Le terme pronostic provient du grec *progignôskein* qui signifie "connaître à l'avance". Ce processus est associé au concept de Condition Based Maintenance (CBM) [22], et plus spécifiquement à la maintenance prévisionnelle. Il a pour but de prévoir l'état futur d'un système, d'un équipement ou d'un composant élémentaire (mécanique, hydraulique, électronique, ...).

On trouve dans la littérature plusieurs définitions qui se rapportent au pronostic, mais deux d'entre elles reviennent principalement :

Définition 2.1. *Le pronostic consiste à calculer une prédiction de l'état d'un composant ou d'un système dans le futur.*

Cette définition reste très générale et assimile le pronostic à une prédiction d'états de croyance du système. Le temps n'est pas nécessairement quantifié. Or, du point de vue de la maintenance, il est nécessaire de prédire la date à partir de laquelle le composant ou le système évolue d'un état opérationnel (en bon fonctionnement) vers un état non opérationnel, c'est-à-dire en fonctionnement anormal.

Une deuxième définition, énoncée ci-dessous, tient compte des objectifs de la maintenance préventive.

Définition 2.2. *Le pronostic est la capacité de prédire la durée de vie résiduelle ("Remaining Useful Life") ou RUL de composants ou systèmes en service.*

La RUL d'un système correspond au temps restant avant que le système ne puisse plus réaliser avec succès ses fonctions requises et doit être remplacé (figure 2.1) [38].

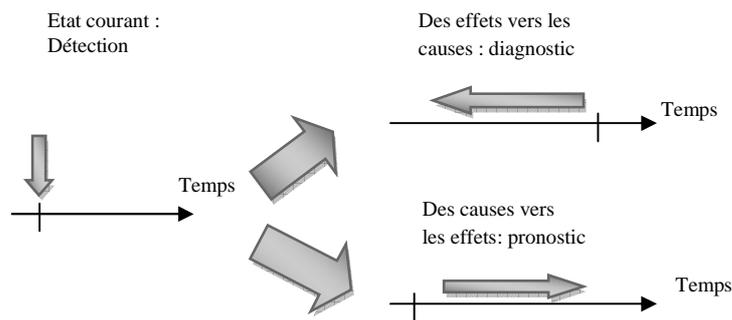


Figure 2.1 *Détection, diagnostic et pronostic : les aspects phénoménologiques.*

Le pronostic est généralement efficace pour les modes de défaillance présentant des caractéristiques de détérioration connues. Une défaillance doit donc être dénie en termes de paramètres ou de descripteurs surveillés. Ceci suppose évidemment que la situation courante peut être saisie (pratiquement, le pronostic est la synthèse d'un procédé de détection et des données mesurées du système). Par ailleurs, le concept même de défaillance implique que le pronostic doit être associé à un degré d'acceptabilité (un système doit exécuter une fonction exigée). Nous considérons ainsi que le pronostic devrait être basé sur les critères d'évaluation, dont les limites dépendent du système lui-même et des objectifs d'exécution. Tout ceci nous conduit à retenir la définition proposée par la norme ISO04 [12] :

Définition 2.3. *Le pronostic est l'estimation de la durée de fonctionnement avant défaillance et du risque d'existence ou d'apparition ultérieure d'un ou de plusieurs modes de défaillance.*

2.2.1 Processus de pronostic

Notons que le diagnostic est, par nature, rétrospectif et focalisé sur des données existant à un instant donné. Cependant, le pronostic est focalisé sur l'avenir et, de ce fait, doit prendre en compte les aspects suivants ISO04 [12] :

- les modes de défaillance existants et les taux de détérioration,
- les critères de déclenchement de futurs modes de défaillance,

- le rôle des modes de défaillance existants dans le déclenchement de futurs modes de défaillance,
- L'influence entre les modes de défaillance existants et les modes futurs et leurs taux de détérioration,
- la sensibilité à la détection et aux modifications des modes de défaillance existants et futurs du fait des techniques de surveillance actuelles,
- la conception et les changements de stratégies de surveillance afin de s'adapter à tous les éléments ci-dessus,
- l'effet des actions de maintenance et/ou des conditions d'exploitation,
- les conditions ou les hypothèses dans lesquelles les pronostics restent valables.

Parmi les processus précédemment développés, le processus de pronostic est l'élément clé de la stratégie de maintenance prévisionnelle car il supporte la notion d'anticipation qui caractérise une approche proactive. Paradoxalement, il est aussi identifié comme le moins avancé en terme de formalisation et d'intégration: «Le pronostic de l'évolution de la dégradation est aujourd'hui une science très difficile qui reste encore un domaine très balbutiant [04](figure 2.2).

L'intégration du processus de pronostic dans le système de maintenance prévisionnelle, sa modélisation et la formalisation de son principe de construction constituent aussi bien des problématiques académiques [32] qu'industrielles [37].

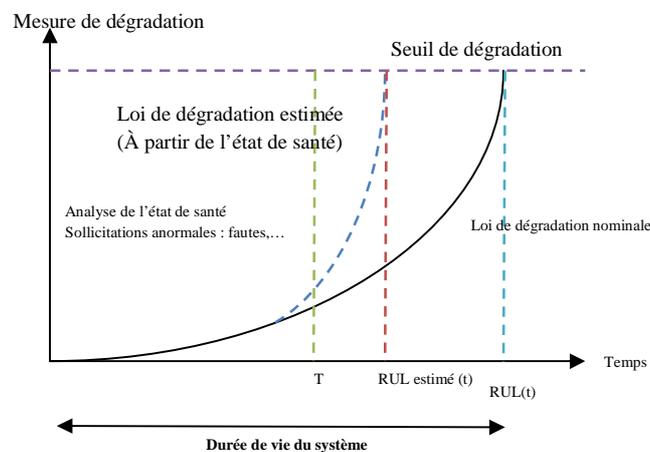


Figure 2.2 Loi de vieillissement pour le pronostic.

2.3 Les indicateurs de pronostic

Le résultat de la prédiction est la future valeur estimée du processus (y). Analysant cette valeur, les incertitudes inhérentes au processus de prédiction doivent être prises en considération (très peu d'informations sur le phénomène étudié, difficultés dans la formalisation).

Ce sont des sources très importantes d'erreurs (e) qui font la différence entre la valeur future réelle mesurée du système (y) et celle prévue précédemment ($e = y - \hat{y}$).

En particulier, l'erreur pourra être considérée :

- par rapport au moment de détection d'un défaut T (lorsque le système atteint la limite de dégradation acceptable $xx\%$ au delà de laquelle la performance est considérée comme insatisfaisante par les industriels),
- par rapport au moment d'apparition de la défaillance (dégradation 100%).

2.3.1 L'erreur moyenne absolue

L'erreur absolue moyenne en pourcentage MAPE (*Mean Absolute Percentage Error*) «équation 2.1 » : est la moyenne des écarts en valeur absolue par rapport aux valeurs observées. C'est donc un pourcentage et par conséquent un indicateur pratique de comparaison.

Le MAPE ne peut s'appliquer qu'à des valeurs strictement positives. Il permet donc de juger si le système de prévision est bon.

La plupart des auteurs se rapportent essentiellement à l'erreur moyenne absolue exprimée en pourcentage - MAPE « équation 2.1 » et également à l'écart type des erreurs (σ) « équation 2.2 » pour juger de la qualité d'un système prédictif [35] [36] [37].

$$MAPE = \sqrt{\frac{1}{N} \sum_{i=1}^N Abs\left(\frac{y_i - \hat{y}_i}{y_i}\right)} \quad 2.1$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad 2.2$$

Où σ : l'écart type des erreurs

Y valeur future réelle de paramètre surveillé, y valeur future prédite de paramètre surveillé. N le nombre des échantillons.

MAPE tient compte de l'ordre de grandeur de l'erreur de prédiction et évite ainsi les cas "non déterminées" du point de vue mathématique, survenu suite à l'apparition d'une valeur d'erreur trop proche de 0 [14].

Par contre, certains auteurs conviennent que la fonction coût associée aux erreurs de prédictions, devrait être employée dans l'évaluation d'une méthode de prédiction et que le

MAPE devrait être utilisé alternativement (le cas de la fonction coût est linéaire dans le pourcentage, mais pas dans l'erreur absolue).

2.3.2 L'erreur quadratique moyenne

les études récentes et l'expérience des opérateurs du système indiquent que la fonction coût dans le problème de prédiction est clairement non linéaire, et que les grandes erreurs peuvent avoir des conséquences désastreuses. Pour cette raison, des mesures basées sur l'erreur quadratique moyenne RMSE (*Root Mean Square Error*) « équation 2.3 » sont parfois suggérées, car elles pénalisent les erreurs énormes peuvent ainsi être considérées comme mieux adaptées.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \tag{2.3}$$

2.3.3 Temps restant avant défaillance

La principale métrique de pronostic recherchée est le temps restant avant défaillance (Time To Failure : TTF). En outre, une mesure de confiance peut être construite pour indiquer le degré de certitude du temps avant défaillance prédit. Par extension, et en considérant que les utilisateurs peuvent être intéressés par l'évaluation du système au regard de limites de performance, le RUL et la confiance peuvent être généralisés : sur la figure 2.3, TT_{xx} est le temps restant pour dépasser la limite de performance $Perf_{xx}$, et $Conf_{xx}T$ est la confiance avec laquelle l'indication $TT_{xx}/T > T$ peut être prise.

Une mesure de confiance (confidence) peut être construite pour indiquer le degré de certitude du temps prévu jusqu'à la défaillance, tout en considérant que les praticiens peuvent être intéressés par l'évaluation du système par rapport à n'importe quelle limite d'exécution.

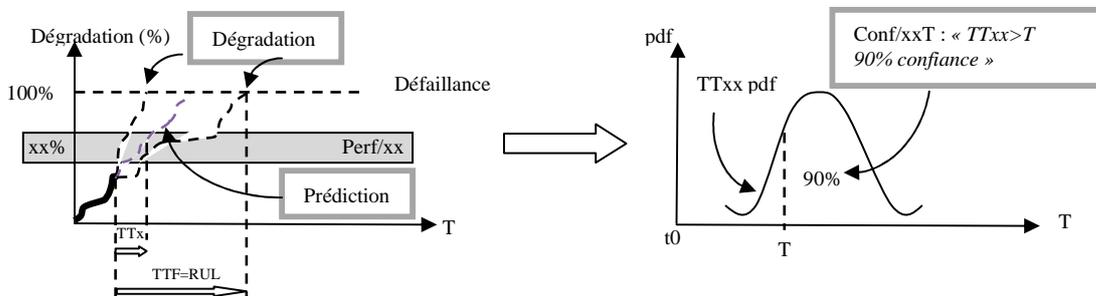


Figure 2.3 TT_{xx} et confiance d'après [14]

2.3.4 L'exactitude

L'exactitude (accuracy) mesure la proximité de la date de défaillance prévue avec la date de défaillance réelle. Le calcul de cette métrique représente un point critique dans le processus de pronostic. La question qui se pose tout d'abord est si la prédiction est "assez bonne". La réponse dépend fortement de la rigidité des critères d'évaluation imposés.

Pour cela, le calcul de cette grandeur s'appuie sur l'existence de données historiques sur plusieurs composants tombés en panne suite à des sollicitations subies tout au long d'une période de temps connue, ce qui n'est pas toujours possible.

Si un ensemble de N systèmes sont tombés en panne (avec les pronostics associés), l'exactitude est définie par « équation 2.4 » [38]

$$\text{Exactitude}(t_p) = \frac{1}{N} \sum_{i=1}^N e^{-\frac{D_i}{D_0}} \quad 2.4$$

Telle que :

- $D_i = |t_p(i) - t_0(i)|$ est la distance entre les instants réels et prévus de défaillance,
- D_0 est un facteur de normalisation, une constante dont la valeur est basée sur l'importance de la valeur réelle dans l'application,
- N est le nombre d'expériences

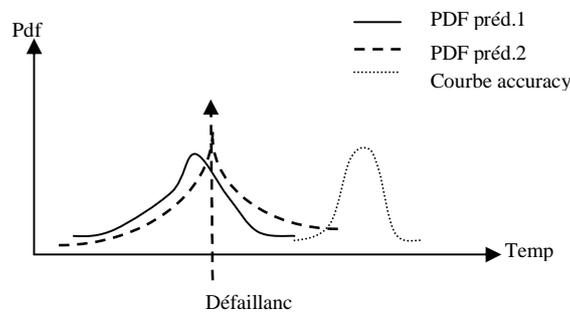


Figure 2.4 L'exactitude d'après [38]

La fonction exponentielle $e^{-\frac{D_i}{D_0}}$ est employée ici pour donner une courbe lisse monotoniquement décroissante. La valeur de $\frac{D_i}{D_0}$ représente la décroissance de l'exactitude. Autrement dit, l'exactitude est grande (proche de 1) quand la valeur prévue est identique à la valeur réelle et diminue quand la valeur prévue dévie de la valeur réelle. La fonction exponentielle a également le taux de décroissance le plus grand quand D_0 est proche de 0. Une illustration de cette métrique est proposée en figure 2.4.

2.3.5 La Précision

La précision est une mesure de dispersion des prédictions. Elle permet d'évaluer comment les valeurs prédites sont groupées autour de l'intervalle dans lequel survient la défaillance (figure 2.4). La précision dépend fortement du niveau de confiance et de la distribution des prédictions. L'équation de précision est la suivante :

$$\text{précision}(t_p) = \left(\frac{1}{N} \sum_{i=1}^N e^{\frac{D_i}{D_0}} \right) e^{\frac{\delta_i}{D_0}} \quad 2.5$$

Avec :

- $E_i = t_{fail}(i) - t_{pred}(i)$,
- $\bar{E} = \frac{1}{N} \sum_{i=1}^N E_i$, $\delta^2 = \frac{1}{N} \sum_{i=1}^N (E_i - \bar{E})^2$,
- δ_0 et R_0 Des facteurs de normalisation, et R_i l'intervalle de confiance de la prédiction pour l'expérimentation i .

2.3.6 Opportunité

L'Opportunité où (timeliness) est la position relative de la fonction de densité de probabilité (pdf : probability density function) du modèle de pronostic par rapport à l'apparition de l'événement de défaillance. Cette mesure évolue à mesure que les données sont disponibles et permet de juger du moment adéquat pour effectuer la maintenance (figure 2.5) [22]. P. Cochetoux préconise de définir des limites au plus tôt et au plus tard au delà desquelles la valeur prédite doit être considérée comme inacceptable d'un point de vue performance.

Ces deux limites sont la conséquence du fait que l'erreur de prédiction n'est pas systématiquement centrée par rapport à zéro (où l'erreur est dénie comme la différence entre la vie restante réelle et la vie restante estimée). Par exemple, si la prédiction est trop tôt, l'alarme résultante sollicite trop tôt l'intervention pour vérifier le potentiel d'apparition d'une défaillance, pour surveiller les diverses variables de processus et pour effectuer une modalité de reprise.

Dans l'autre cas, si la défaillance est prévue trop tard, cette erreur réduit le temps disponible pour évaluer la situation et réagir en conséquence. La situation devient catastrophique quand la défaillance se produit avant qu'une prévision ne soit faite.

Par conséquent, il est dans la plupart des situations préférable d'avoir une concentration positive des erreurs (prévisions tôt), plutôt que négative (prévisions en retard). Naturellement, il est nécessaire de définir des limites d'acceptabilité pour le timeliness (trop tôt, ou trop tard) (figure 2.5). Toute prédiction en dehors des limites est considérée comme inadaptée.

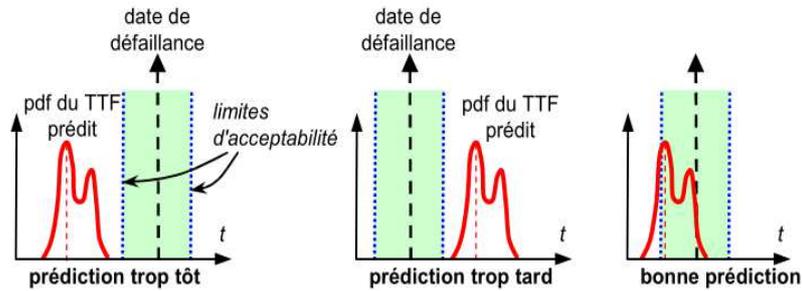


Figure 2.5 Mesure de performance de système de pronostic «Timeliness »

2.4 Les Méthodes de pronostic industriel

De nombreux outils et méthodes de pronostic de défaillance ont été proposés durant la dernière décennie. Les méthodes de pronostic diffèrent généralement par le type d'application considérée, alors que les outils mis en œuvre dépendent principalement de la nature des données et connaissances disponibles pour construire un modèle de comportement du système réel incluant le phénomène de la dégradation. Aussi, ces méthodes et outils peuvent être regroupés dans un nombre limité d'approches.

Le premier classement des approches de pronostic a été proposé par [45]. Les auteurs suggèrent une classification pyramidale à trois niveaux des approches de pronostic (figure 2.6) : les approches basées sur l'expérience, les approches guidées par les données, et les approches basées sur les modèles.

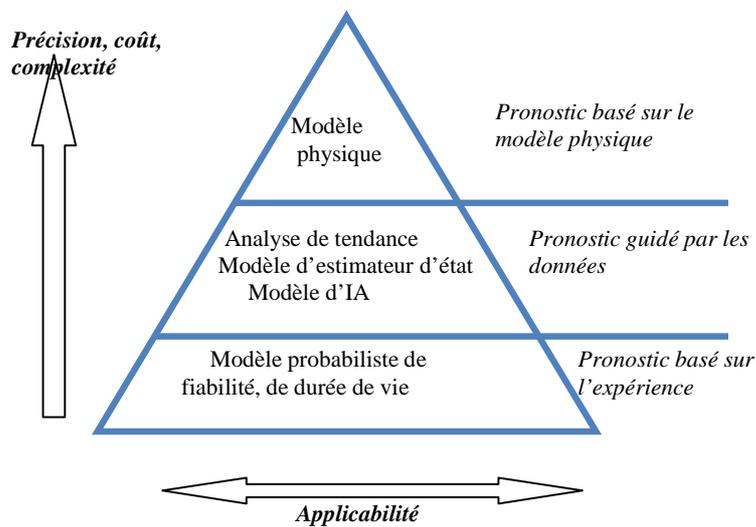


Figure 2.6 Taxonomie des approches de pronostic

- *Approches basées sur les modèles*: elles s'appuient sur des modèles issus des lois physiques ainsi que des modèles de dégradation construits par expérimentation permettant d'obtenir une représentation mathématique du mécanisme de dégradation.
- *Approches basées sur les données*: elles s'appuient sur l'utilisation de données directement mesurées sur le système ou d'indicateurs calculés à partir de celles-ci. Ces données peuvent représenter l'état actuel issu du système de surveillance et/ou l'état passé issu d'historiques de mesures obtenues lors d'utilisations passées d'équipements similaires. Dans ce cadre on peut distinguer des pronostics :
 - ✓ par analyse de tendance : prédiction de l'évolution future d'un indicateur par l'ajustement d'un modèle à l'aide des données représentatives de l'état courant (prédiction par série temporelle, régression...),
 - ✓ par apprentissage : ces approches sont supportées par des modèles issus de l'IA qui utilisent les historiques de la dégradation lors d'une phase d'apprentissage afin d'ajuster les paramètres du modèle. Les réseaux de neurones et les outils Neuro-flou constituent les principaux outils pour ces approches,
 - ✓ Par estimation d'état : ces approches sont utilisées notamment lorsqu'un diagnostic par reconnaissance de formes est mis en œuvre au préalable. Dans ce cas le pronostic consiste à prédire l'évolution de la trajectoire de la forme (indicateur de dégradation) principalement par l'intermédiaire d'un filtre de Kalman.
- *Approches basées sur l'expérience* : elles s'appuient sur des modèles fiabilistes ou des Modèles stochastiques de la dégradation qui sont construits sur le retour d'expérience Composé des temps de passage d'un état à un autre (temps de défaillance pour les modèles fiabilistes) ou d'avis d'experts. Ils sont principalement utilisés lorsqu'il est trop difficile d'élaborer un modèle physique et qu'aucun dispositif de suivi de l'état de dégradation n'est opérationnel [22].

Cependant, Vasile [14] a proposé une classification originale (figure 2.7) en utilisant comme critère le contenu des entrées du pronostic : données, informations et connaissances. Dans ce cadre, les données représentent le contenu le plus simple, sans aucun sens propre si elles ne sont pas interprétées (ex. 90°C), les informations sont des ensembles de données structurées dans un contexte incluant une signification (température actuelle d'un processeur:90°C) et la connaissance est la manipulation d'informations laissant percevoir une capacité d'action.

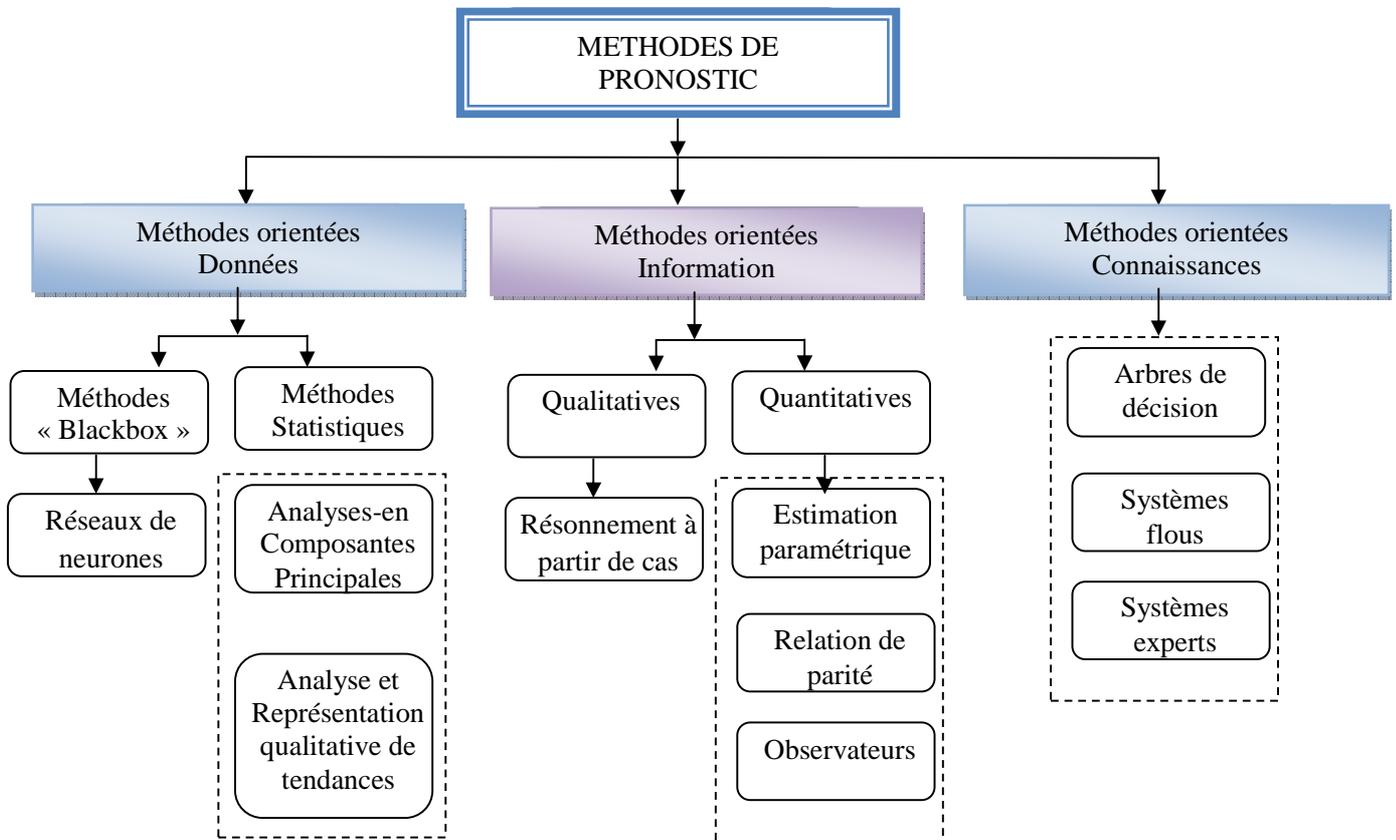


Figure 2.7 La classification des méthodes de pronostic

2.4.1 Méthodes basées sur la connaissance

Dans les travaux récents nous retrouvons les méthodes basées sur la connaissance. Les connaissances offrent aux utilisateurs des informations supplémentaires liées à la vie restante jusqu'à la défaillance du processus étudié.

Les méthodes de pronostic basé sur l'expérience se situent en bas de la pyramide. Ces méthodes constituent la seule alternative possible dans la situation où il n'y a aucune connaissance disponible sur la nature physique du système et ses composants nombre de capteurs est limité et n'est généralement pas conçu pour résoudre un problème de pronostic. Une approche de pronostic basée sur l'expérience repose sur un modèle de pronostic obtenu à partir d'une connaissance de surface. Cette forme de modèle est la plus simple et ne nécessite que l'*historique* des défaillances ou les recommandations de conception des composants dans des conditions opérationnelles similaires afin de déterminer la probabilité de défaillance à un moment donné dans le futur.

Les approches basées sur la connaissance ont la capacité de modéliser les systèmes identifiés avec une déformation minimale de la réalité. Ceci est dû au fait que les données de surveillance sont liées au contexte. Le langage utilisé est très proche du langage normal et une compréhension physique du système est plus facile.

2.4.1.1 Raisonnement à partir de cas

Un exemple de pronostic basé sur l'expérience est le raisonnement à partir de cas [46]. Le RaPC est un outil de calcul dont la seule source de connaissance est une mémoire des cas importants passés. Quand un problème est résolu, sa solution est retenue pour résoudre des problèmes similaires. Cela permet alors d'éviter de répéter les erreurs commises dans le passé. Dans cette méthode les efforts d'acquisition de connaissances sont réduits, l'efficacité de la résolution des problèmes augmente à mesure qu'ils sont utilisés et elle peut s'adapter aux changements d'environnement. En revanche, il n'existe pas de méthode systématique permettant de choisir les paramètres, la difficulté tient justement à cette structure de cas et aux informations qu'elle doit contenir. En effet, l'extraction des connaissances et leurs représentations sont primordiales dans ce type d'application

2.4.1.2 Les arbres de décision

Les arbres de décision peuvent apprendre les cas non rencontrés au cours de l'utilisation. Le problème se complique en revanche lorsque l'on y ajoute des contraintes temporelles qui elles, ne sont pas fréquemment explicitées. Elles nécessitent donc une très bonne expertise du système et de ces dysfonctionnements.

2.4.1.3 Les systèmes flous

Les systèmes flous nécessitent des connaissances exprimées par des règles. Ce sont des méthodes causales recommandées si nous n'avons pas d'information quantitative mais que des règles de dépendance décrivant la propagation des défauts. Un système flou peut être ajusté automatiquement et les modèles mathématiques ne sont pas requis. L'information antérieure sur les règles peut être utilisée parce que l'interprétation et l'implémentation sont simples. Les SF constituent aussi le seul cadre dans lequel on peut traiter des imprécisions et des incertitudes et qui autorise également le traitement de certaines incomplétudes.

En revanche :

- les règles doivent être disponibles,
- ils n'ont pas la capacité d'apprendre,
- l'adaptation aux changements de l'environnement est difficile,

- aucune méthode formelle pour l'ajustement de la base de connaissance créée initialement n'est disponible.

Les systèmes experts sont des méthodes transparentes qui raisonnent dans des conditions incertaines (floues) qui par contre, explicitent les décisions prises. La facilité de développement de ce type d'outil est une conséquence du fait qu'ils n'exigent pas beaucoup de détails liés au système. Les bases de connaissances sur lesquelles ils s'appuient sont très spécialisées, nécessitent une abondance d'expériences et sont difficiles à actualiser.

2.4.2 Méthodes guidées par les données

Le pronostic guidé par les données s'appuie sur le constat suivant : les mesures (entrées/sorties) constituent souvent la plus forte et la plus sûre source d'information pour comprendre les phénomènes de dégradation. Sa force réside dans l'aptitude à apprendre (par les exemples) et à capturer les relations subtiles entre données, même si ces relations sont inconnues ou difficile à décrire.

Elles sont utilisées dans le cas où des observations en ligne sont disponibles mais où on ne possède toujours pas de connaissance physique sur la nature du système. Les modèles de pronostic considérés dans ces approches reposent alors sur des estimateurs (estimateur d'état, estimateur de paramètre) qui sont utilisés en ligne pour évaluer l'état de dégradation du système. Les paramètres considérés ici sont des caractéristiques mesurables du système qui peuvent être utilisées comme indicateur de vieillissement pour prédire les défaillances et donner des estimations du RUL [24].

L'inconvénient principal de ces approches est que leur efficacité est grandement dépendante de la quantité et de la qualité des données opérationnelles de système.

2.4.2.1 Estimation d'état et de paramètres

Les méthodes de pronostic par estimation d'état ou de paramètres sont utilisées pour prédire le comportement d'un paramètre révélateur de l'état de dégradation du système. Il s'agit de minimiser l'écart entre un modèle de référence de ce paramètre significatif et les mesures obtenues à partir des capteurs à un instant donné. Le modèle de pronostic utilisé n'a pas de connaissance physique du système mais il est le résultat d'une analyse mathématique du comportement. Des techniques classiques d'automatique pour la prédiction de mesures sur des paramètres en fonction des mesures précédentes sont utilisées. Une fonction d'extrapolation sur la mesure est donc nécessaire. Par exemple, la fonction d'extrapolation tangentielle calcule une prédiction de mesure sur un paramètre à l'instant $n + 1$ en fonction de la mesure de ce paramètre à l'instant n :

$$f(n + 1) = f(n) + \dot{f}(n)t + \frac{1}{2}\ddot{f}(n)t^2 \tag{2.8}$$

Où $f(n)$ est une mesure à la date n d'un paramètre du système à pronostiquer et t est la période entre deux instants de mesure. Cette prédiction repose seulement sur la mesure à l'instant précédent et sur une équation dynamique. La méthode d'estimation permet d'obtenir un pronostic adaptatif. Le filtre de Kalman peut être utilisé pour minimiser l'erreur entre le modèle de pronostic et les données observées pour prédire le comportement futur d'un paramètre du système.

Récemment, les méthodes à base d'estimateurs d'état ont obtenu des résultats satisfaisants sur des fours à induction, des moteurs continus ou des boîtes de vitesse [62] (Figure 2.8).

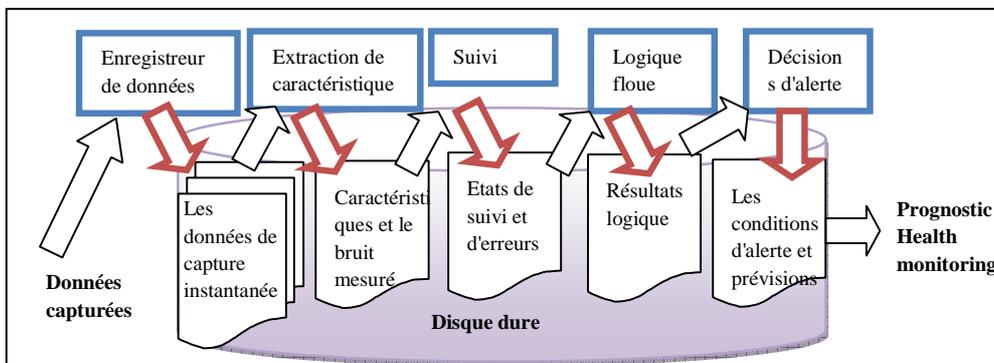


Figure 2.8 Procédure d'exécution d'un algorithme de pronostic [5]

2.4.2.2 Outils d'intelligence artificielle

Lorsque les seules informations disponibles sur le système sont les variables mesurables et que l'on ne peut pas utiliser la redondance physique, la technique habituelle est d'apprendre le comportement du système à l'aide de l'historique des données : ce sont les données d'apprentissage. On part du principe qu'une même cause aura toujours les mêmes effets. Ce sont des systèmes du type "boîte noire" qui a pour principal avantage d'utiliser les données, sans aucune considération physique. Leur force réside dans l'aptitude à apprendre (par les exemples) et à capturer les relations subtiles entre données, même si ces relations sont inconnues ou difficile à décrire.

Apprentissage statistique L'idée principale des techniques d'apprentissage statistiques (classification statistique telle que l'analyse en composantes principales ou les machines à vecteurs de support) est de déterminer des sous-espaces dans un espace vectoriel dans lequel sont représentés des mesures du système [38]. Lors de la phase d'estimation, une nouvelle mesure d'un paramètre est observée et selon le sous espace dans lequel elle se situe, un mode de dégradation peut être associé au paramètre considéré.

Les réseaux de neurones La fonction d'estimation est réalisée par un réseau de neurones qui modélise la progression d'un paramètre jusqu'à la défaillance du système [24]. L'apprentissage va consister à ajuster les poids et les seuils des neurones afin d'obtenir les sorties prédites désirées pour chaque paramètre du système (méthode de rétro-propagation du gradient). Le réseau de neurones est ensuite utilisé pour prédire l'évolution de la dégradation de ces paramètres dans des conditions opérationnelles similaires.

Un exemple d'approche de pronostic à base de données appliquée à un moteur de turbine à gaz est développé dans [14]. Cette approche repose sur des paramètres spécifiques du système et sur un mode pour l'efficacité du compresseur. Une technique probabiliste est développée. Elle utilise l'information disponible sur la manière dont évolue la dégradation des paramètres dans le temps afin d'évaluer la sévérité du changement de distribution de ces paramètres et de projeter leur état futur.

Les applications récentes exploitent des réseaux de neurones sous toutes leurs formes : réseau de neurones à ondelettes dynamiques [5], réseau de neurones à fonctions de base radiales [8] ou réseau Neuro-Flou [14]. L'intégration de la logique floue dans la construction du modèle permet de définir la structure du modèle et d'assurer la convergence de l'apprentissage des paramètres.

Ces méthodes d'apprentissage automatique ont donné lieu à de multiples applications :

- pronostic de fissures des engrenages [67],
- pronostic de l'état d'un système non-linéaire (four à gaz) [5] (Figure 2.9),
- pronostic de défaillances de roulements à billes [60].

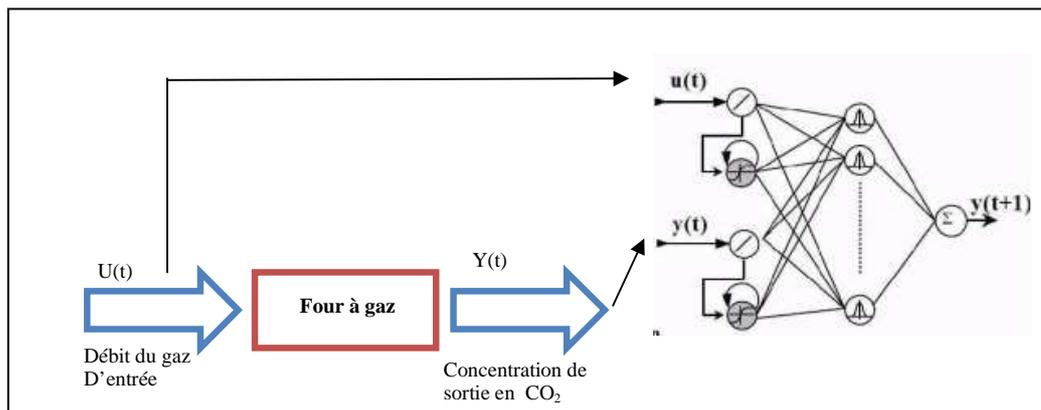


Figure 2.9 Pronostic d'un four à gaz par le réseau de neurones [5]

2.4.3 Méthodes orienté informations

Les méthodes basées sur l'information, dans cette proposition, se basent sur deux catégories de techniques : quantitatives et qualitatives. Nous considérons dans la catégorie des méthodes basées sur l'information qualitative comme candidats potentiels les outils du raisonnement à partir de cas (RaPC) et la reconnaissance de formes.

Leurs points forts sont :

- les efforts d'acquisition de connaissances sont réduits,
- ils sont relativement faciles à maintenir,
- l'efficacité de la résolution des problèmes augmente à mesure qu'ils sont utilisés,
- ils permettent d'utiliser des données existantes dans les bases de données,
- ils peuvent s'adapter aux changements d'environnement.

En revanche :

- la définition précise des classes entre lesquelles va s'opérer la décision,
- il n'existe pas de méthode systématique permettant de choisir les paramètres,
- l'information existante sur les différents modes de fonctionnement d'un système est toujours incomplète,
- la difficulté tient justement à cette structure de cas et aux informations qu'elle doit contenir. En effet, l'extraction des connaissances et leurs représentations sont primordiales dans ce type d'application.

Les techniques quantitatives (espace de parité, estimation paramétrique, observateurs) sont en fait des techniques statistiques, habituellement employées dans des méthodes basées sur les modèles "classiques"(utilisent des seuils pour détecter la présence des défauts).

La force de cette catégorie de techniques réside dans leur capacité à prendre en compte dans la conception de l'architecture de système, les défauts multiples et à apporter des informations sur l'importance des déviations. Ces techniques nécessitent cependant d'avoir un système physique excité en permanence, ce qui en limite l'applicabilité industrielle. Les processus sont habituellement décrits dans le mode stationnaire et sont très coûteux, ce qui est impossible pour les systèmes dangereux. Habituellement leur fonctionnement hors ligne empêche les traitements en temps réel. La modélisation des perturbations qui peuvent engendrer des erreurs dans le modèle parce que l'adaptabilité de ces approches aux changements du processus n'existe pas.

2.5 Pronostiquer la défaillance

Dans (Léger et Morel)[14] puis (Muller)[22] les auteurs proposent une architecture d'un système intégré de maintenance prévisionnelle (SIMP) (voire la Figure 2.10).

Cette architecture est constituée uniquement de trois processus : un processus de surveillance et diagnostic, un processus de pronostic et un processus d'aide à la décision. La spécificité de cette architecture réside dans le bouclage explicite entre les processus de pronostic et d'aide à la décision. Ainsi le processus de pronostic est, pour l'aide à la décision, le support d'évaluation des alternatives de maintenance ou de conduite, permettant de prévenir la défaillance.

Cette vision sur le pronostic est basée sur les résultats de Muller et al. [22]. En ce sens la figure 2.12 présente une extension formalisée de la figure 2.11 pour modéliser précisément l'environnement du SIMP constitué des trois processus principaux de surveillance et de diagnostic, de pronostic et d'aide à la décision

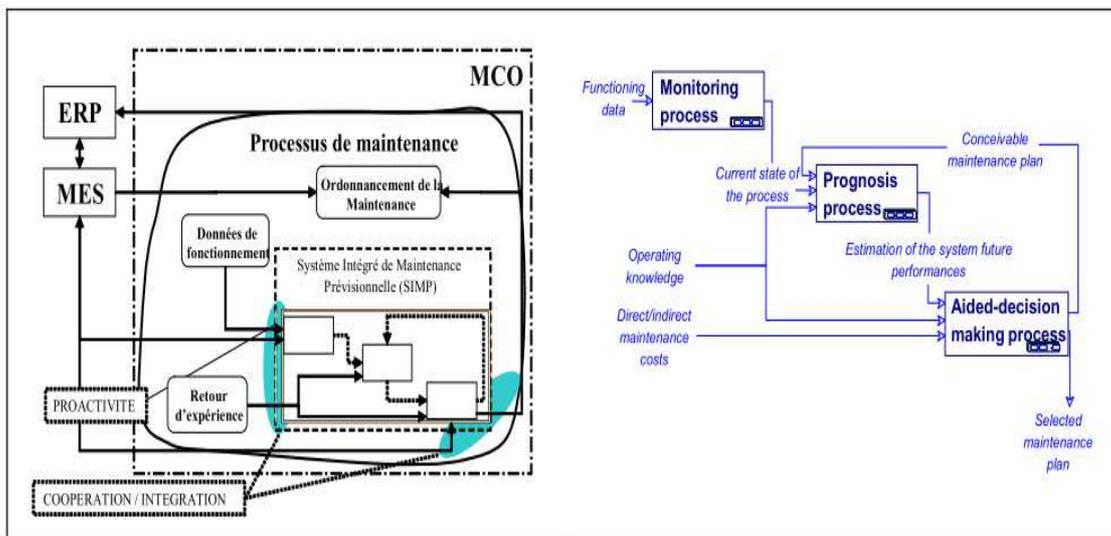


Figure 2.10 Système intégré de maintenance prévisionnelle [22]

A cela s'ajoute plusieurs processus situés à différents niveaux de modélisation de l'entreprise. Au niveau système/process (opérationnel), on retrouve le processus « Acquérir et prétraiter » qui permet d'obtenir des données directement issues du système, puis de les traiter et de les rendre disponibles. Au niveau tactique et opérationnel, les processus « Ordonnancer la production » et « Gérer la maintenance » sont supportés par des outils de type MES, GMAO. Enfin le processus « Gérer l'entreprise » se situe au niveau stratégique (ERP).

Les processus du SIMP trouvent leur correspondance dans l'architecture d'OSA-CBM. Ainsi, le processus « Surveiller et diagnostiquer » correspond à la couche « Condition Monitoring » et une partie de la couche « Health Assessment ». Le processus « Pronostiquer la défaillance » correspond à la seconde partie de la couche « Health Assessment » et à la couche « Pronostics ». Enfin, le processus « Aider à la décision » correspond à la couche « Décision support ». Les autres couches d'OSA-CBM correspondent à des processus de l'environnement du SIMP.

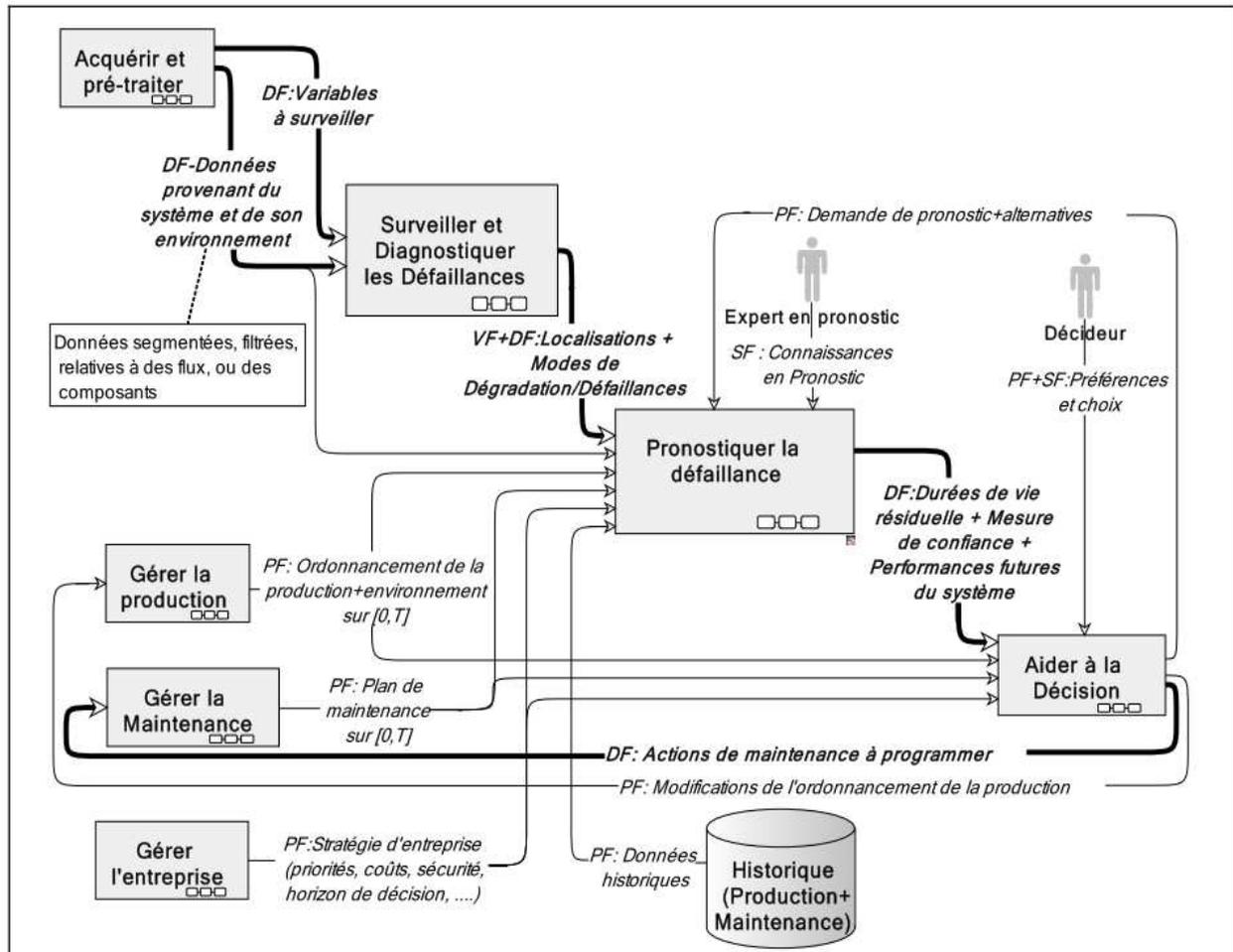


Figure 2.11 Environnement de processus de pronostic [22]

2.5.1 Les sous-processus de pronostic

Le processus de pronostic se décompose en quatre sous-processus présentés Figure 2.12. Le sous-processus « Piloter le pronostic » contrôle l'activité des trois autres sous-processus : « Initialiser l'état et les performances », « Projeter dans le futur » et « Calculer les durées de vie résiduelle ». Ces trois sous-processus enchaînés constituent les étapes nécessaires pour obtenir un pronostic.

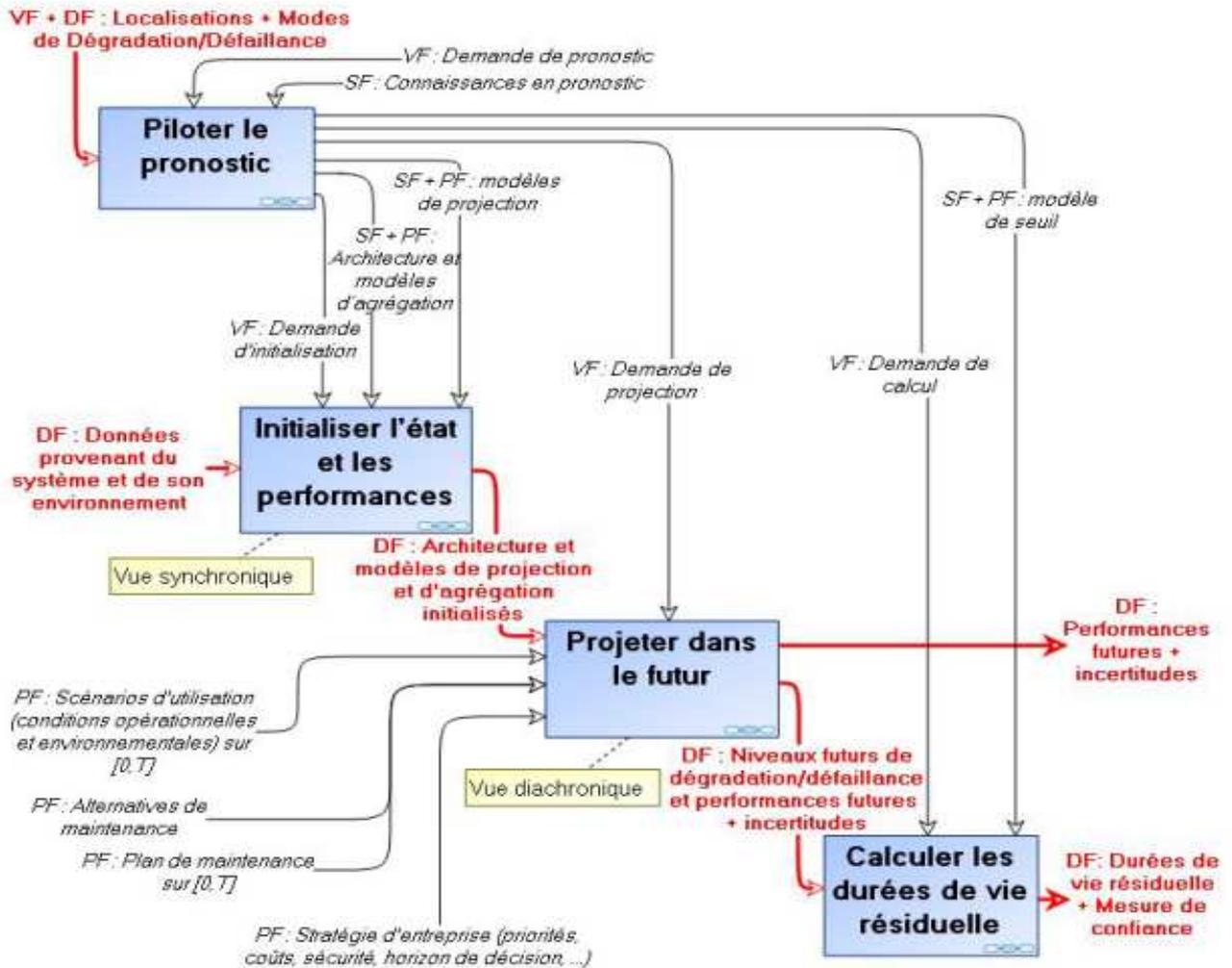


Figure 2.12 Décomposition du processus de pronostic [22]

2.5.1.1 Piloter le pronostic

Le sous-processus « Piloter le pronostic » a pour rôle de contrôler l'activité des trois autres sous-processus. Contrôler signifie d'une part coordonner l'activité des autres sous-processus et d'autre part leur fournir les modèles appropriés à la situation courante (Savoir Faire), i.e. La localisation des fautes et les modes de dégradation/défaillance courants (résultat du processus de diagnostic).

Ainsi, « Piloter le pronostic » possède 2 types de sortie :

- Des demandes d'action aux autres sous-processus,
- Les modèles nécessaires à leur activité.

2.5.1.2 Initialiser l'état et les performances

La finalité du sous-processus « Initialiser l'état et les performances » est de fournir un point de départ au processus « Projeter dans le futur ». Il permet d'actualiser la situation courante du système et de ses composants avec les dernières données disponibles. Il fournit une vision synchronique du système et de ses composants. La sortie est constituée de l'état courant des composants et des performances actuelles du système et des composants :

- L'état courant des composants est donné par le niveau de chaque mode de dégradation/défaillance avéré.
- Les performances, composant ou système, sont données par des indicateurs permettant de quantifier l'activité du système ou du composant vis-à-vis de sa finalité.

2.5.1.3 Projeter dans le futur

Le sous-processus « Projeter dans le futur » permet de déterminer l'évolution future des dégradations/défaillances et des performances composants et système afin d'avoir une vue évolutif du système. La projection tient compte de l'influence des conditions opérationnelles et environnementales, ainsi que de l'impact des différentes actions du plan de maintenance (scénario et alternatives).

2.5.1.4 Calculer les durées de vie résiduelle

Ce dernier sous-processus permet de calculer les RUL, finalité du processus de pronostic. Une RUL est définie par la date, dans l'intervalle $[0, T]$ à laquelle la valeur de la projection franchit un seuil. Suivant la valeur projetée le seuil n'est pas défini de la même manière. Il existe deux types de seuil, les seuils composants, qui correspondent à des pannes composants, et les seuils fonctionnels, qui correspondent à une perte de finalité. Pour un composant, les valeurs projetées sont les niveaux des modes de dégradation/défaillance et les performances. Les seuils correspondent à la mise à zéro de la norme ISO 13381-1, i.e. une panne composant.

Au niveau fonctionnel, le seuil est défini par rapport à la finalité d'une fonction et correspond à la limite de performance pour laquelle on considère que la finalité n'est plus atteinte (seuil critique).

2.6 Conclusion

De part sa relative jeunesse, le pronostic est encore un processus dont la finalité, les missions et performances ne sont pas encore totalement stabilisés. Tout d'abord, un outil de pronostic doit évidemment permettre la prévision d'un futur état du système[14]. Les interprétations existantes sont fondées sur la notion de défaillance (ou défaut), qui implique que la situation "prédite" est associée à un degré d'acceptabilité. En outre, les industriels veulent non

seulement connaître ce que sera une situation, mais l'évaluer : quelle performance sera atteinte ? Ainsi, nous avons considéré que le pronostic devrait être basé surtout sur les critères d'évaluation et nous avons considéré la définition du pronostic comme association d'une prévision et une évaluation.

Afin de clarifier cette situation, une cartographie de la terminologie a tout d'abord été donnée. L'étude bibliographique menée a permis d'établir un historique de la communauté « pronostic » puis un l'état de l'art qui a fait ressortir une vue majoritairement centrée sur les modèles de projection. Afin d'établir une vision plus globale des travaux de la communauté, une nouvelle classification a donc été proposée en s'appuyant sur l'analyse contextuelle du pronostic synthétisée à travers trois axes représentatifs de:

- L'objectif de la stratégie de maintenance,
- La prise en compte du contexte opérationnel futur,
- La source de la connaissance.

Les systèmes de pronostic par outils de l'IA peuvent donc représenter d'excellents systèmes d'aide à la décision pour l'expert humain. L'une des méthodes de l'IA la plus utilisée est celle des RNAs artificiels et la LF et les systèmes Neuro-Flous. Leur principe de fonctionnement est inspiré du cerveau humain. Les Réseaux Neuro-Flous sont des outils hybrides capables de traiter des informations de type numérique et symbolique.

Selon la problématique de pronostic industriel, l'introduction du temps dans les techniques des systèmes Neuro-Flous est indispensable pour l'amélioration de prédictions en utilisant des données numérique et symbolique.

En ce sens, le chapitre suivant est consacré à la présentation des notions de base des les systèmes Neuro-Flous et de leur application en pronostic et la surveillance dynamique.

Chapitre III

Les Systèmes Neuro-Flous

Chapitre III

Les systèmes Neuro-Flous

3.1 Introduction

Les Systèmes hybrides qui combinent la logique floue, les réseaux neuronaux, les algorithmes génétiques, et les systèmes experts prouvent leur efficacité dans une variété de problèmes de monde réel et dans l'industrie.

Par exemple, pendant que les réseaux neuronaux sont intéressants pour reconnaître des modèles, ils ne peuvent pas expliquer comment ils atteignent leurs décisions. De même pour les systèmes de la logique floue qui peuvent raisonner avec l'information imprécise, sont intéressants pour expliquer leurs décisions mais ne peuvent pas automatiquement acquérir les règles qu'ils utilisent pour prendre ces décisions.

Ces limites ont été une raison derrière la création de systèmes hybrides intelligents où deux ou plus de techniques sont combinées dans une manière à vaincre les limitations d'une seule technique.

Les applications de l'approche de la logique floue montrent les avantages quand le modèle des systèmes est difficile à implémenter. L'intégration des réseaux de neurones et les systèmes d'inférence floue peuvent être formulés en trois principales catégories: coopérative, concurrente et hybride.

Les principaux avantages des techniques floues sont l'approche de la modélisation et la bonne interopérabilité de la description, en employant des règles linguistiques. Cependant, comme il n'y a aucune méthode formelle pour déterminer ses paramètres (ensembles et règles floues), l'exécution d'un système flou peut prendre beaucoup de temps. Dans ce sens, il serait intéressant de disposer d'algorithmes permettant l'apprentissage automatique de ces paramètres.

L'une des méthodes qui permet de répondre à ces exigences est la théorie des réseaux de neurones qui emploie des échantillons (données d'observation) pour l'apprentissage. La combinaison des deux techniques nous donne les systèmes Neuro-Flous.

[47] à montre que les réseaux de neurones peuvent approximer n'importe quel système de règles floues et inversement les réseaux de neurones peuvent être approximés avec un système d'inférence de règles floue. Différents travaux se sont intéressés à l'intégration des RNA et les SIF.

3.2 Les réseaux de Neuro-Flous

Pendant que les réseaux neuronaux sont intéressants pour reconnaître des modèles, ils ne peuvent pas expliquer comme ils atteignent leurs décisions. De même pour les systèmes de la logique floue qui peuvent raisonner avec l'information imprécise, est intéressants pour expliquer leurs décisions mais ne peuvent pas automatiquement acquérir les règles qu'ils utilisent pour prendre ces décisions. Pour cela, une approche a été développée qui est principalement connu comme approche Neuro-Floue (figure 3.1).

3.2.1 Définitions

Définition 1 :

Le système Neuro-flou est un système flou formé par un algorithme d'apprentissage inspiré de la théorie des réseaux de neurones. La technique d'apprentissage opère en fonction de l'information locale et produit uniquement des changements locaux dans le système flou d'origine [63].

Définition 2 :

Un système Neuro-Flou est un réseau de neurones qui est typologiquement équivalent à la structure d'un système flou. Les entrées/sorties du réseau ainsi que les poids sont des nombres réels, mais les nœuds implémentent des opérations spécifiques aux systèmes flous : fuzzyfication, opérateurs flous (conjonction, disjonction), défuzzyfication [64].

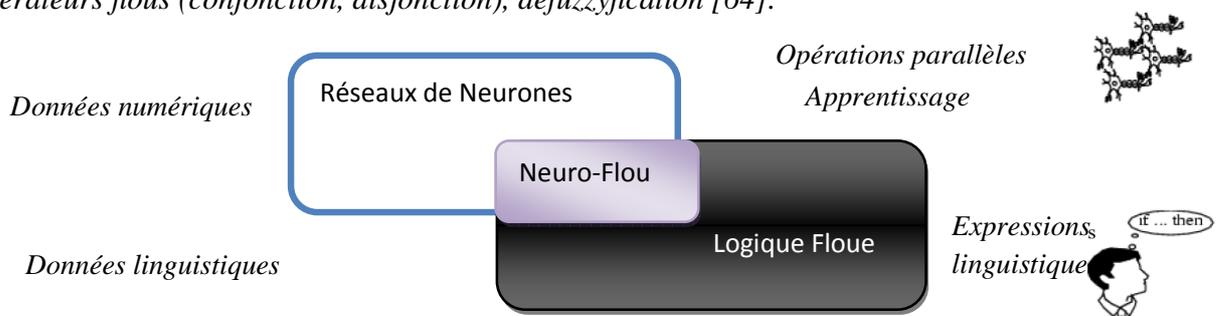


Figure 3.1 Le système Neuro-Flou

Les systèmes Neuro-Flous sont créés afin de synthétiser les avantages et de surmonter les inconvénients des réseaux neuronaux et des systèmes flous. De cette manière, les algorithmes d'apprentissage peuvent être employés pour déterminer les paramètres des systèmes flous. Ceci revient à créer ou améliorer un système flou de manière automatique, au moyen des méthodes spécifiques aux réseaux neuronaux.

Un aspect important est que le système reste toujours interprétable en termes de règles floues, vu qu'il est basé sur un système flou.

Afin de résumer l'apport du Neuro-Flou, le Tableau 3.1 regroupe les avantages et les inconvénients de la logique floue et des réseaux de neurones.

Réseaux de neurones	Logique floue
Avantages	
<ul style="list-style-type: none"> ▪ Le modèle mathématique non requis. ▪ Aucune connaissance basée sur les règles. ▪ Plusieurs algorithmes d'apprentissage sont disponibles. 	<ul style="list-style-type: none"> ▪ Le modèle mathématique non requis. ▪ La connaissance antérieure sur les règles peut être utilisée. ▪ Une interprétation et implémentation simple.
Inconvénients	
<ul style="list-style-type: none"> ▪ Boîte noire (manque de traçabilité). ▪ L'adaptation aux environnements différents est difficile et le réapprentissage est souvent obligatoire (sauf pour RBF). ▪ La connaissance antérieure ne peut pas être employée (apprentissage à partir de zéro). ▪ Aucune garantie sur la convergence de l'apprentissage. 	<ul style="list-style-type: none"> ▪ Les règles doivent être disponibles. ▪ Ne peut pas apprendre. ▪ Adaptation difficile au changement de l'environnement. ▪ Aucune méthode formelle pour l'ajustement.

Tableau 3.1 Comparaison entre la logique floue et les réseaux de neurones

3.3 Méthodes et combinaisons Neuro-Floues

Diverses associations de ces deux approches ont été développées depuis 1988 et sont le plus souvent orientées vers la commande de systèmes complexe et les problèmes de classification. Il existe ainsi trois méthodes Neuro-floues : *coopérative*, *concurrente* et *hybride*.

3.3.1 Le système Neuro-Flou coopératif

L'une des premières méthodes Neuro-Floues est le codage du système d'inférences floue sous la forme d'un réseau de neurones multicouches dans lequel les poids correspondent aux paramètres du système. L'architecture du réseau dépend du type de règle et des méthodes d'inférence, d'agrégation et de défuzzification choisies.

Il utilise des réseaux de neurones et des systèmes flous associés en série ou en parallèle. Plusieurs variantes d'utilisation sont ainsi possibles :

Le réseau de neurones fonctionne en amont du système flou (figure 3.2). Les variantes d'entrées du système flou sont déterminées à partir des sorties du RNA (dans le cas où elles ne sont pas mesurables directement) ou encore un RNA effectue une tâche de classification ou d'approximation, suivie d'un système flou d'aide à la décision. Un réseau de neurones qui fonctionne en aval du système flou, dans le but d'ajuster les sorties d'un système de commande floue à de nouvelles connaissances obtenues, les variables de sorties étant les erreurs sur les variables de sortie du système flou.

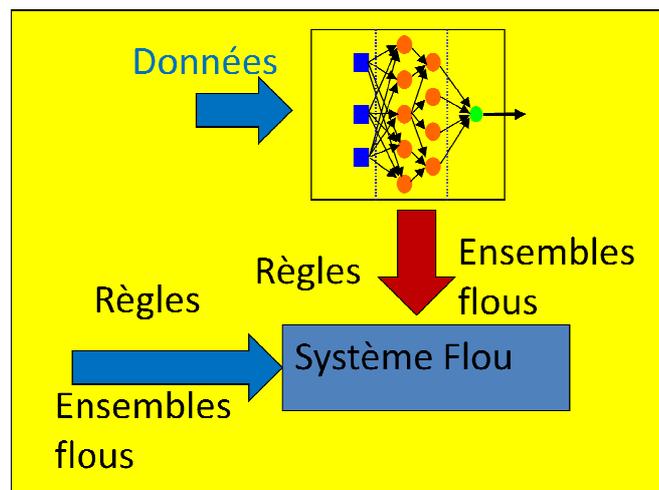


Figure 3.2 Les systèmes Neuro-Flous coopératifs

Deuxième variante est le système en série, qui serait utilisé si la sortie n'est pas convenable pour une relation direct à l'entrée du système flou. Les systèmes post-traitement existent aussi et dans lesquels la sortie d'un système flou n'est pas convenable par rapport direct aux systèmes externes, et par conséquent un réseau neurone fournit une interface qui exécute une projection topographique qui ne pourrait pas être porté facilement (figure 3.2).

On trouve aussi un système Neuro-flou parallèle qui fait une coopération entre les réseaux de neurones et les systèmes flous en parallèle en même temps.

Le modèle du système Neuro-flou coopératif peut être considéré comme un pré- processeur avec des mécanismes d'apprentissage des RNA détermine les fonctions d'appartenance du système d'inférence floue ou les règles floues quand les paramètres de SIF (système d'inférence floue) sont déterminés.

3.3.2 Le système Neuro-Flou concurrent

Il utilise des réseaux de neurones et des systèmes flous associés en série ou en parallèle. Plusieurs variantes d'utilisation sont ainsi possibles :

Le RNA fonctionne en amont du système flou. Les variantes d'entrées du système flou sont déterminées à partir des sorties du réseau de neurones (dans le cas où elles ne sont pas mesurables directement) ou encore un réseau de neurones effectue une tâche de classification ou de reconnaissance de formes, suivie d'un système flou d'aide à la décision (figure 3.3(a)).

Un RNA qui fonctionne en aval du système flou, dans le but d'ajuster les sorties d'un système de commande floue à de nouvelles connaissances obtenues, les variables de sorties étant les erreurs sur les variables de sortie du système flou (figure 3.3(b)).

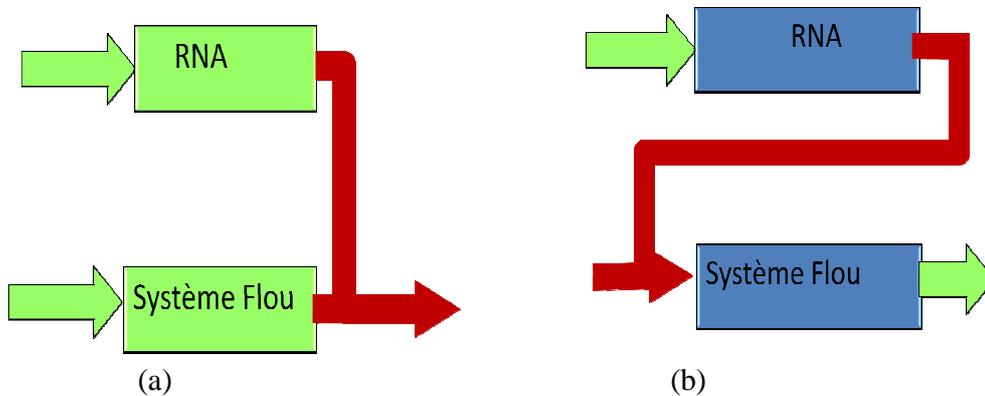


Figure 3.3 Les systèmes Neuro-Flous concurrents

3.3.3 Le système Neuro-Flou hybrides

Une architecture à 3 modèles flous est proposée dans [3] :

La règle J ($j=1,2,\dots,j$) :

$$SI x_1 \text{ est } A_1^j \text{ et } \dots \text{ et } x_n \text{ est } A_n^j, \text{ ALORS } y_1 \text{ est } f_1^j \text{ et } \dots \text{ et } y_m \text{ est } f_m^j \quad (3.1)$$

$$\text{Ou } f_k^j = \begin{cases} B_k^j & \text{(type I)} \\ \theta_k^j & \text{(type II)} \\ b_{0k}^j + b_{1k}^j x_1 + \dots + b_{nk}^j x_n & \text{(type III)} \end{cases} \quad (3.2)$$

Avec x_i ($i=1,2,\dots,n$), y_k ($k=1,2,\dots,n$) respectivement les variables d'entrées et de sortie

A_1^j Les ensembles flous d'entrée ; B_k^j, θ_k^j et $b_{0k}^j + b_{1k}^j x_1 + \dots + b_{nk}^j x_n$ représentant respectivement, des ensembles flous de sortie, des composants singleton et des combinaisons linéaires des variables d'entrées.

La Figure 3.4 montre les différentes architectures des systèmes Neuro-flous. FALCON (a) et GARIC (b) interprètent la règle floue de type I avec une structure à 5 couches, NEFCLASS (c) interprète la règle floue de type II avec une structure à 3 couches et ANFIS (d) interprète la règle floue de type III avec une structure à 6 couches.

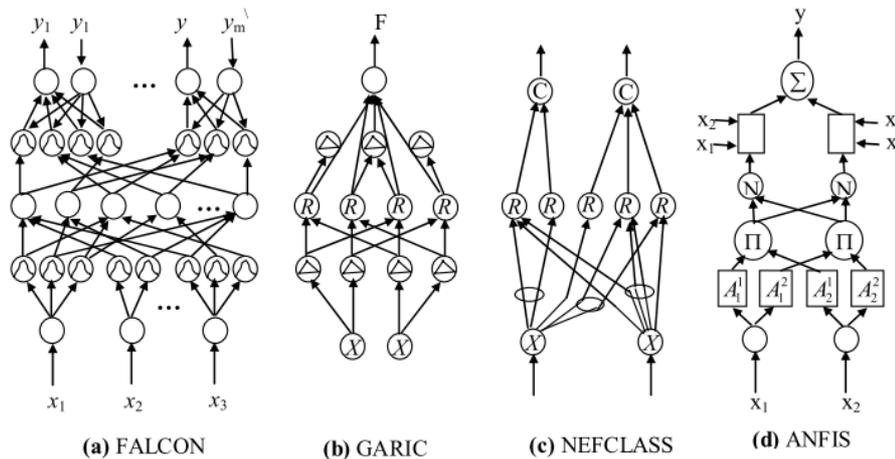


Figure 3.4 Les Différentes architectures des systèmes Neuro-flous

• FALCON et GARIC

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

• NEFCLASS

Modèle utilisé généralement en classification, il est constitué de 3 couches : une couche d'entrée avec les fonctions d'appartenance, une couche cachée représentée par des règles et une couche de sortie définissant les classes [63][64][65] (Figure 3.4).

Ce modèle est facile à mettre en application, il évite l'étape de défuzzification, tout en étant précis dans le résultat final, avec une rapidité bien supérieure aux autres architectures.

- **NEFCON**

Il s'agit de modèle formé de 3 couches. Une couche cachée formée par des règles, une couche d'entrée incluant les nœuds d'entrée avec les sous-ensembles flous d'antécédentes et une couche de sortie avec un nœud de sortie et les sous-ensembles des conséquents. L'élaboration des règles est similaire à l'architecture NEFCLASS, avec une différence en sortie. Cette architecture est généralement utilisée en approximation de fonctions et en contrôle flou.

Le processus d'apprentissage du NEFCON peut être divisé en deux phases. La première phase consiste à trouver les règles de base initiales. Si les connaissances antérieures ne sont pas disponibles, les règles de base seront apprises avec difficulté. Et si cette règle est définie par un expert l'algorithme les complète. Dans la seconde phase, les règles de base sont optimisées par modification des sous-ensembles flous des règles. Les deux phases utilisent l'erreur floue, cette erreur peut être trouvée avec la différence entre la sortie désirée et celle obtenue [64].

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

ANFIS représente un système à inférence floue mis en application dans le cadre des réseaux adaptatifs. Il utilise la procédure d'apprentissage hybride. Cette architecture (Figure 3.4 d) affine les règles floues obtenues par des experts humains pour décrire le comportement d'entrée-sortie d'un système complexe. Ce modèle donne de très bons résultats en poursuite de trajectoire, approximation non linéaire, commande dynamique et traitement du signal.

Les points forts de l'ANFIS consistent dans le mécanisme d'inférence distributif et l'algorithme d'apprentissage adaptatif. Par contre les règles sont non interprétables et l'apprentissage se fait hors ligne.

- **NEFPROX (Neuro Fuzzy function apPROXimator)**

Modèle obtenu par l'association des deux architectures NEFCLASS et NEFCON, il est utilisé dans différentes applications comme la classification et l'approximation de fonctions [63]. NEFCLASS utilise un algorithme supervisé pour définir les règles floues, le NEFCON utilise un algorithme d'apprentissage non supervisé avec le calcul de l'erreur de sortie. Les deux modèles emploient la rétro propagation afin de définir les sous-ensembles flous.

Comparé au modèle ANFIS, NEFPROX est beaucoup plus rapide, mais ANFIS donne de meilleurs résultats en approximation. Le NEFPROX est le premier système interprétable et lisible, dédié à l'approximation de fonction. Néanmoins, ses résultats en classification restent moins bons que ceux donnés par le NEFCLASS.

- **HyFIS (Système Neuro-flou hybride)**

Il a été introduit par (Kim et Kasabov) en 1999 [16]. Il est constitué de par deux parties : Un module d'analyse flou pour l'extraction des règles floues par les données des entrées par l'utilisation de la méthode de Wang 1994. Un module connexionniste qui implémente et règle les règles floues à travers l'application de l'algorithme du rétro propagation (voir la figure 3.5).

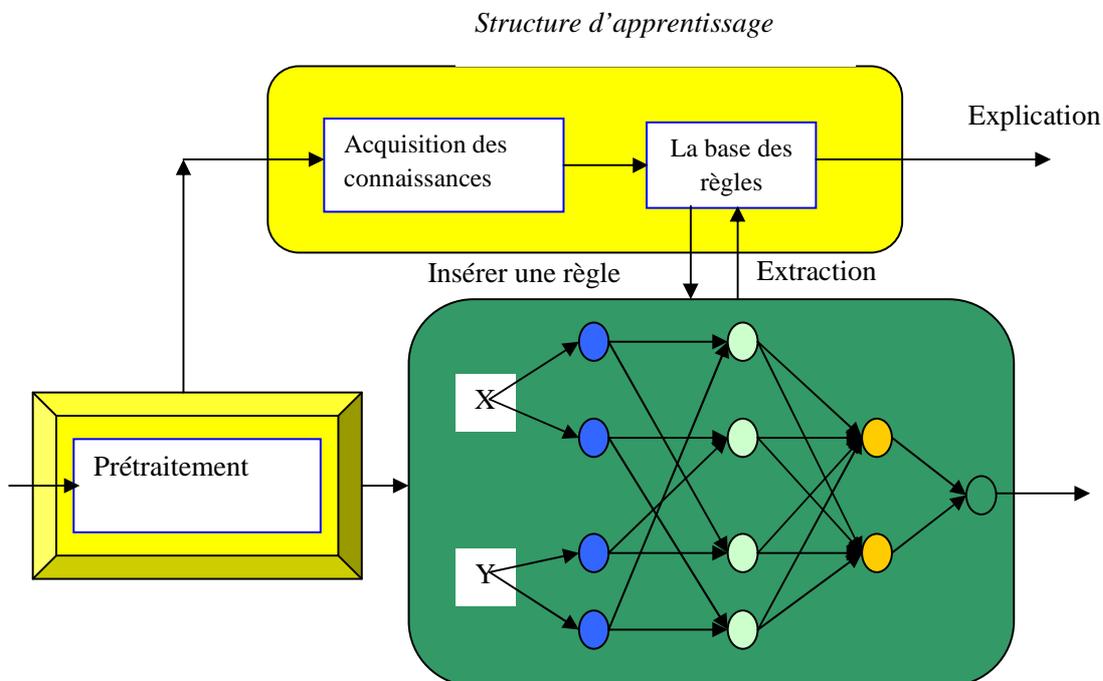


Figure 3.5 Le schéma bloc du diagramme de HyFIS

- **NEFDIAG (Neuro-Fuzzy DIAGnosis)**

Il a été introduit par (Mahdaoui et al [80]), NEFDIAG peut être représenté comme un type spécial de perceptron flou, à trois couches utilisé pour classifier des défaillances.

NEFDIAG fait son apprentissage par un ensemble de formes, telle que chaque forme soit affectée (classée) vers une des classes prédéfinies (figure 3.6). NEFDIAG génère les règles floues par un parcours de données et optimise ensuite les règles par apprentissage des paramètres des

sous ensembles flous qui sont utilisés pour partitionner les données « caractéristiques » des formes à classer et les paramètres des données. NEFDIAG représente un classificateur flou F_r avec un ensemble de classes $C = \{c_1, c_2, \dots, c_m\}$. Après l'apprentissage, NEFDIAG peut être utilisé pour classifier une nouvelle observation, le système peut être représenté sous forme de règles floues :

Si symptôme est A_1 symptôme $_2$ est A_2 symptôme $_3$ est A_3 symptôme est A_n **Alors** la forme $(x_1, x_2, x_3, \dots, x_n)$ est à la classe mode de défaillance $_1$.

Des corrections sont ensuite apportées aux différents paramètres du système (les poids, les seuils et les paramètres des fonctions d'appartenance). L'utilisateur définit le nombre initial des fonctions d'appartenance pour partitionner les domaines des données d'entrées et spécifie le nombre k , nombre maximum de neurones des règles qui seront créés dans la couche cachée.

Initialisation : pour chaque donnée issue des capteurs, il y a une unité d'entrée, et pour chaque mode de défaillance il y a une unité de sortie. Pour chaque unité d'entrée une partition floue initiale est spécifiée (ex. Nombre des fonctions d'appartenance triangulaire).

Apprentissage des règles : Le système NEFDIAG peut démarrer avec une base de connaissance partielle des formes, et raffiner durant l'apprentissage. La règle sera créée par la recherche (pour une forme donnée f) la combinaison des fonctions d'appartenance telle que chaque entrée produit la plus grande fonction d'appartenance. Si cette combinaison n'est pas identique pour les règles existantes dans la base des règles et le nombre de règles n'est pas maximum, alors une règle sera créée et ajoutée à la base des règles.

Apprentissage des FA : pour l'apprentissage des fonctions d'appartenance, une simple rétro propagation sera utilisée. Une décision sera créée qui dépend de l'erreur de sortie pour chaque unité de règles. Chaque règle change ses fonctions d'appartenance par le changement de leurs supports. NEFDIAG 2.00 est un pro logiciel pour la conception, l'apprentissage, équilibrage et test d'un système Neuro-flou pour la surveillance dynamique d'un système de production.

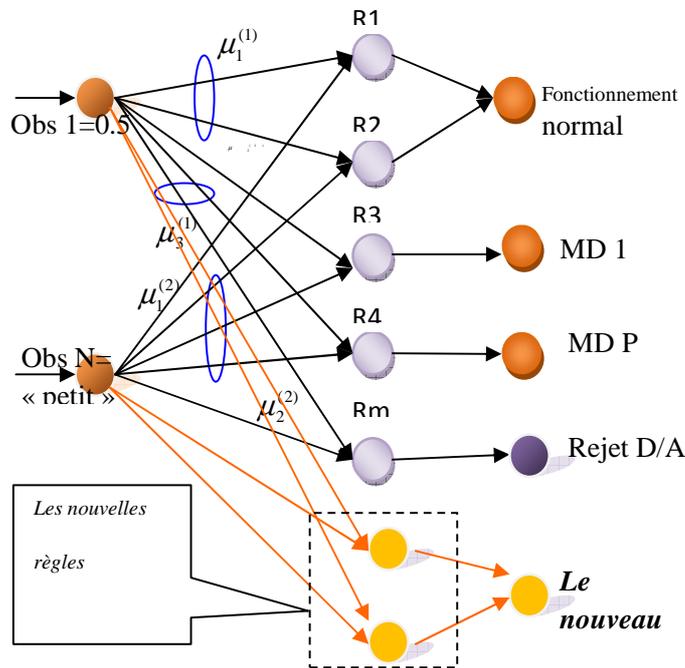


Figure 3.6 L'architecture de NEFDIAG

La forme des règles de NEFDIAG est :

Si symptôme 1 **est** A1 symptôme2 **est** A2 symptôme3 **est** A3 symptôme **est** An

Alors la forme (x1, x2, x3,..., xn) est à la classe Mode de défaillance1.

Après avoir détecté la dégradation le système NEFDIAG fait sont intervention par le réglage de la variable consterné.

3.3.4 Apprentissage des systèmes Neuro-Flous

Dans la majorité des cas rencontrés la stratégie d'apprentissage repose sur une adaptation de l'algorithme de rétro propagation du gradient. On trouve cependant quelques autres exemples [13] utilisant une méthode de perturbation aléatoire des poids du réseau et ne gardent que les modifications qui améliorent les performances. Si une telle approche ne garantit absolument pas la convergence vers un système optimal, elle possède l'avantage de n'imposer aucune contrainte sur la structure du réseau.

La rétro-propagation du gradient de l'erreur est utilisée pour ajuster les poids et les biais du réseau afin de minimiser l'erreur quadratique entre la sortie du réseau et la sortie réelle. A chaque couple entrée/sortie, une erreur est calculée, le gradient, ou pente, de l'erreur est déterminé. Ensuite les poids et les biais sont modifiés en ligne sur le réseau. On réitère ces calculs jusqu'à l'obtention du critère d'arrêt.

3.4 Les avantages des systèmes Neuro-Flous

3.4.1 La rapidité de calcul

Les systèmes Neuro-flous sont très rapides parce que l'évaluation d'un système compétent implique typiquement des opérations de la comparaison simples et un nombre limité de calculs de la surface linéaire. Cette accélération est analogue à l'algorithme de l'alpha bêta de la théorie des jeux. Les arbres de décision sont très rapides, et la comparaison des paramètres est faite d'une manière floue pas comme la méthode classique et enfin la base des règles floue sera repartie et que chaque règle fonctionne d'une manière autonome et en même temps collectif.

3.4.2 La Flexibilité

Un système Neuro-flou peut traiter des problèmes complexes avec beaucoup de variables de l'entrée. Au lieu d'adapter une architecture interne fixe, l'architecture d'un système Neuro-Flou peut grandir dynamiquement et efficacement en réponse à la complexité des données d'apprentissage. La structure d'un système Neuro-Flou représente efficacement le rapport entre les entrées du problème et les sorties (tableau 3.2).

3.4.3 Généralisation des connaissances

Les systèmes d'apprentissage ont souvent des difficultés quand il y a un manque de données historiques pour former, ou bien les données contiennent trop de bruit. Les Systèmes Neuro-Flous peuvent compenser ces problèmes pendant la phase de l'apprentissage en changeant (adapter) leur structure interne. Souvent les règles sont de la forme: plus les conditions de l'entrée satisfaisante, plus des résultats satisfaisants.

Les réseaux de neurones artificiels RNA	Système d'inférence flou SIF	Système Neuro-Flou SNF
La base de règle ne peut être utilisée	La base de règle peut être utilisée	La base de règle peut être utilisée.
L'apprentissage peut démarrer de zéro	Pas d'apprentissage (utilise la connaissance linguistique).	L'apprentissage peut être démarré de zéro ou avec une base de règles initiale
Boite noire	Interprétable (la règle if-then)	Une interprétation et implémentation simples
Complexité des algorithmes d'apprentissage	Universel et implémentation simple	Le modèle mathématique non requis.
Difficulté pour extraire les connaissances	Les connaissances doivent être disponibles	Simple pour extraire les connaissances a partir des données (qualitatif, quantitatif).

Tableau 3.2 Les avantages des systèmes Neuro-flous.

3.5 Les applications des systèmes Neuro-Flous

Les premières applications de RNA flous à produits du consommateur sont parues sur le marché (Japonais et Coréen) en 1991. Quelques exemples incluent des appareils à conditionner de l'air, électrique, ventilateurs électriques, thermo pots électriques, appareils de chauffage du ventilateur du kérosène, fours à micro-ondes, réfrigérateurs, machines à laver, et des photocopieurs.

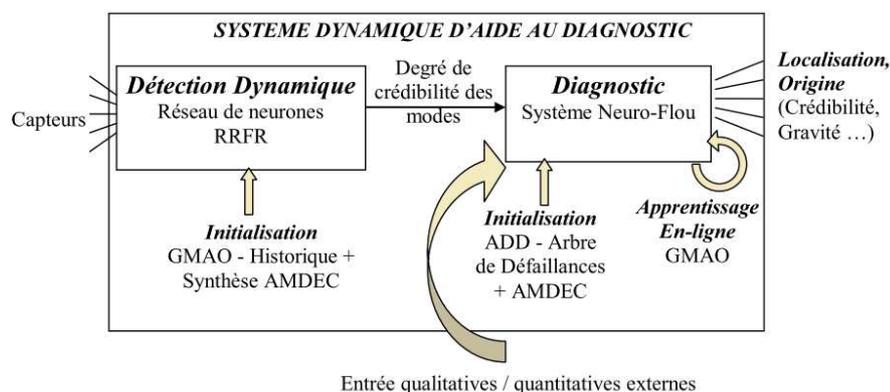


Figure 3.7 Schéma du système d'aide au diagnostic.

N. Palluat dans le travail de thèse de [15], propose l'utilisation de techniques Neuro-Floues dynamique pour une aide dynamique au diagnostic (figure 3.7):

En fonctionnement, le système de détection scrute en permanence le système surveillé. Lorsqu'une panne ou une défaillance se produit ou lorsque l'une des deux est susceptible de se produire, une alarme est déclenchée et un diagnostic est lancé, le système de détection n'arrêtant pas sa tâche. En utilisant les informations provenant du système de détection, le système d'aide au diagnostic propose à l'opérateur les causes possibles de ce problème, ainsi que les interprétations floues de ces causes. Cette méthode comprend plusieurs étapes :

- ✓ L'acquisition des informations pertinentes du système
- ✓ Application du système de détection basé sur les réseaux de neurones dynamiques
- ✓ Application du système de diagnostic basé sur un réseau Neuro-Flou

L'outil est formé de deux parties, une partie neuronale récurrente (RRFR) pour la détection dynamique et une deuxième partie Neuro-Floue pour l'aide au diagnostic.

La limite de cette application est l'intégration des systèmes Neuro-Flous dans un AdD donc a ce niveau on parle d'une surveillance quasi intelligente parce que les modes de défaillances sont reconnues à l'avance.

3.6 Conclusion

Dans ce chapitre, nous avons présenté les systèmes Neuro-Flous leurs architectures et les différents modèles et types d'apprentissage associés. Quelques uns de ces modèles sont des approximateurs universels, d'autres sont des contrôleurs et d'autres des classificateurs, nous avons ensuite détailler l'étude sur les systèmes Neuro-Flous.

En effet, ces systèmes combinent les capacités d'apprentissage et de généralisation des réseaux de neurones, avec la formalisation de règles imprécises ou entachées d'incertitudes correspondant à une information humaine approximative ou à une information partielle ou partiellement erronée, comme c'est souvent le cas dans les applications industrielles.

Les systèmes Neuro-flous présentés dans ce chapitre peuvent offrir des solutions très intéressantes dans des applications de reconnaissance des formes ou approximation de fonctions mais ne peuvent en aucun cas être appliqués sur des données où le temps joue un rôle déterminant dans la résolution du problème.

1. Comment ces systèmes Neuro-Flous statiques peuvent être adaptés pour être dynamiques?
2. Quelles sont les architectures de réseaux temporels qui existent en littérature ?
3. Comment est mené l'apprentissage pour la prise en compte de la dimension temporelle des données ?

Nous essaierons de répondre à ces questions à travers le chapitre suivant consacré à la représentation du temps dans les systèmes Neuro-Flous.

Nous utilisons ces connaissances pour obtenir un système Neuro-Flou Temporel, flexible qui utilise le système Neuro-flou hybride comme un outil de détection et diagnostic dans un système de production choisi, et utilise les techniques de reconnaissance de formes floue qui sont compatibles avec les systèmes Neuro-Flous Temporels.

Chapitre IV

Les Systèmes Neuro-Flous Temporels: *Outil pour la surveillance Dynamique*

Chapitre IV

Les Systèmes Neuro-Flous Temporels : Outil pour la Surveillance Dynamique

4.1 Introduction

Nous avons démontré dans le chapitre précédent que parmi l'ensemble des représentations des systèmes Neuro-Flous, les Systèmes Neuro-Flous Temporels (SNFT) sont les plus performants pour des applications de surveillance dynamique. Les systèmes Neuro-Flous dynamiques se montrent favorables aux trois types d'applications des SNFT en surveillance dynamique, à savoir la reconnaissance de séquences temporelles, la prédiction temporelle et la reproduction de séquences temporelles. Les réseaux Neuro-Flous récurrents sont donc les seuls à posséder une mémoire dynamique interne.

Les techniques d'apprentissage de ces réseaux sont souvent très lourdes à mettre en œuvre. Une application de surveillance dynamique avec ce type de SNFT peut être très compliquée à cause de cette phase d'apprentissage très complexe et surtout coûteuse en temps de calcul.

Dans ce chapitre, un algorithme de retro propagation de gradient pour un apprentissage optimal des Systèmes Neuro-Flous Temporels est proposé avec une nouvelle structure de réseau de neurones d'Elman appelée (système Neuro-Flou d'Elman).

4.2 Réseaux de neurones récurrent Elman

La représentation du temps dans les réseaux de neurones représente une caractéristique essentielle dans la perspective d'une surveillance industrielle dynamique. La prise en compte de l'aspect temporel des données par les RNAs nécessite certaines modifications architecturales des modèles neuronaux statiques.

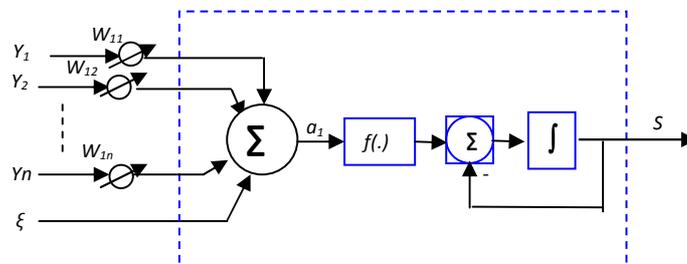


Figure 4.1 Représentation du comportement dynamique d'un neurone récurrent

Il existe dans la littérature deux façons distinctes d'aborder le temps par les réseaux de neurones. Dans la première, le temps est représenté comme un mécanisme externe au réseau de neurones. Des retards servent à mémoriser les données d'entrée pendant une certaine durée t_i . Par contre, dans la deuxième le réseau de neurones est capable de traiter le temps sans aucun mécanisme externe. Cette représentation est appelée représentation interne selon Chappelier [06] et représentation dynamique selon Elman [38] (Figure 4.1).

Dans l'architecture proposée par Elman (1990), les unités de la couche cachée sont dupliées sur une couche appelée couche de contexte avec un poids unitaire. L'apprentissage s'effectue par l'algorithme de rétro propagation et ne concerne que les poids de propagation avant (figure 4.2).

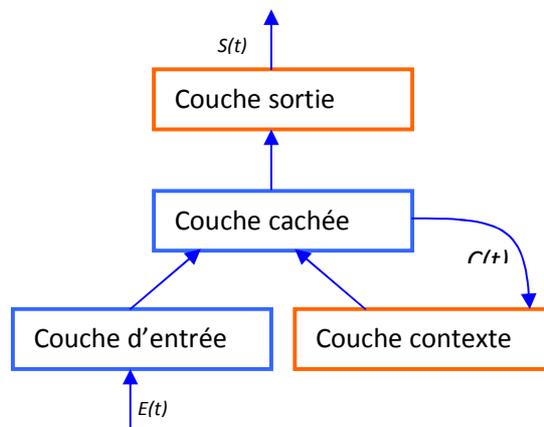


Figure 4.2 Architecture récurrente d'Elman.

4.2.1 La structure du système Neuro-Flou Elman

Dans cette section, nous présentons notre méthode pour décrire un système non-linéaire en utilisant des règles floues dans la forme des équations l'espace d'état linéaires comme des conséquences.

Le réseau de Elman à des neurones non linéaire, dans sa couche cachée récurrente et un neurone linéaire dans sa couche de sortie. Ce type de réseau peut aussi approximer n'importe qu'elle type de fonction pourvu que la couche cachée ait assez de neurones.

Puisque ce réseau peut enregistrer de l'information pour une référence future, il est capable d'apprendre des associations aussi bien temporelles que spatiales.

4.2.1.1 les règles floues temporelles

Les règles de type Takagi-Sugeno sont utilisés récemment pour la structuration systèmes Neuro-Flous [9]. Cette règle d'inférence significative fournit une manière analytique de l'analyse de la stabilité des systèmes de contrôle flou. Si nous combinons les contrôleurs de Takagi-Sugeno

avec le système de contrôle et d'utiliser de l'espace d'état pour décrire l'ensemble du système dédié [10], nous pouvons obtenir un autre type de règles pour décrire les systèmes non linéaires comme ci-dessous:

Règle r: Si X_1 est $T_{x_1}^r$ et et X_n est $T_{x_n}^r$ et

U_1 est $T_{u_1}^r$ et et U_M est $T_{u_M}^r$

alors $\dot{X} = A^r X + B^r U$

Telle que :

$X = [x_1 \ x_2 \ \dots \dots \dots \ x_n]^T$ est le vecteur d'état du système non-linéaire,

$U = [u_1 \ u_2 \ \dots \dots \dots \ u_n]^T$ Est le vecteur d'entrée du système et N, M la dimension du système.

$T_{x_1}^r, T_{u_1}^r$ Sont des termes linguistiques (ensembles flous) qui définissent par les conditions pour x_i et u_i tel que :

$A^r = (a_{ij}^r)_{N \times N}$ est une matrice de $N \times N$ et $B^r = (b_{ij}^r)_{N \times M}$ matrice de $N \times M$

Après l'introduction du temps dans la règle précédente on peut déduire la nouvelle règle comme suit :

Si $X_1(t)$ est $T_{x_1}^r$ et et $X_n(t)$ est $T_{x_n}^r$ et

$U_1(t)$ est $T_{u_1}^r$ et est $U_M(t)$ est $T_{u_M}^r$

alors $X(t + 1) = A^r X(t) + B^r U(t)$

Où $X = [x_1(t) \ x_2(t) \ \dots \dots \ x_n(t)]^T$

Les règles de type Takagi-Sugeno et le système de contrôle, sous la forme ci-dessus sont des règles conformes pour simuler ou identifier tous les systèmes non linéaires, avec ou sans contrôleurs. L'antécédent d'une telle règle définit un sous-espace flou de X et U, et la conséquence qui identifie le système linéaire peut être considéré non linéaire comme dans ce sous-espace. Dans les deux formes, la sortie du système est toujours définie par:

$$Y = CX \text{ (ou } Y(t) = CX(t)) \tag{4.1}$$

Tel que $C = (c_{ij})_{P \times N}$ X est une matrice de $P \times N$ et P est la dimension de sortie du vecteur Y.

La procédure d'inférence floue est spécifiée comme ci-dessous :
Premièrement, nous utilisons la multiplication comme opérateur et pour le poids de la règle .

Le poids de la règle r:

$$f_r = \prod_{i=1}^N \mu_{T_{x_i}^r} [x_i(t)] \cdot \prod_{i=1}^M \mu_{T_{u_i}^r} [u_i(t)] \quad (4.2)$$

Où $\mu_{T_{x_i}^r}$ et $\mu_{T_{u_i}^r}$ sont des fonctions d'appartenance de $T_{x_i}^r$ et $T_{u_i}^r$ respectivement

Après la normalisation de la règle, faire une sommation de règles R est le nombre total des règles)

$$S = \sum_{r=1, n}^R f_r, h_r = f_r / S \quad (4.3)$$

Où S est la sommation des poids de toutes les règles, et h est le poids normalisé de la règle r. après la défuzzification, nous avons :

$$X^r(t + 1) = A^r X(t) + B^r U(t),$$

$$\begin{aligned} X(t + 1) &= \sum_{r=1}^R h_r X^r(t + 1) \\ &= \sum_{r=1}^R h_r [A^r X(t) + B^r U(t)] \\ &= (\sum_{r=1}^R h_r A^r) X(t) + (\sum_{r=1}^R h_r B^r) U(t) \\ &= AX(t) + BU(t) \end{aligned} \quad (4.4)$$

$$\text{Où } A = (\sum_{r=1}^R h_r A^r), B = (\sum_{r=1}^R h_r B^r)$$

Pour cela, les règles floues temporelles peuvent être utilisées dans d'autres applications comme la reconnaissance de formes dynamique.

4.2.1.2 Architectures de Réseau Neuro-Flou Temporel d'Elman

Le système Neuro-Flou Temporel d'Elman (NFTE) est une classe de réseau adaptatif. Il peut être vu comme un réseau de neurones flous bouclé pour lequel chaque couche est un composant d'un système Neuro-Flou et, à ce titre, c'est un approximateur et classificateur universel. Il est ainsi utilisé dans différentes applications de prédictions.

Un système NFTE réalise une approximation non linéaire et de la variable de sortie en décomposant l'espace des entrées en différents espaces flous. Considérons la Figure 4.3

pour décrire l'architecture d'un système NFTE et expliquer brièvement le mécanisme d'inférence d'un tel système.

La figure 4.4 montre la structure du réseau SNFT à cinq couches, dont l'architecture de base provient des réseaux de neurones d'Elman et les architectures Neuro-Flous. Dans ce réseau, les nœuds d'entrée qui reçoivent les entrées de système « Observations » et les nœuds de contexte, qui fait une propagation de la valeur du vecteur d'état à partir de la couche 4 tout les sorties calculer seront retro propager à la couche 1 (la couche d'entrée).

Couche 1 : Chaque nœud dans cette couche calcule le coefficient de la 1^{ième} règle à la somme de toutes les règles :

$$o^{[1]} = e^{-\frac{(u^{[1]} - c^r)^2}{2(s^r)^2}} \quad (4.5)$$

Où les C_R et S_R sont respectivement le centre (moyenne) et variation de $u [1]$ de terme linguistique correspondante de l'entrée $u [2]$ dans la règle r .

Les paramètres utilisés dans les fonctions d'activations (classiquement, il s'agit de gaussiennes ou de sigmoïdes) sont appelés paramètres "prémises ". La valeur alors obtenue $\mu (A1)_X$ représente le degré d'appartenance de la valeur $A1$ à l'ensemble X , la figure 4.3 illustre deux types de d'ensemble flou triangulaire et trapèze.

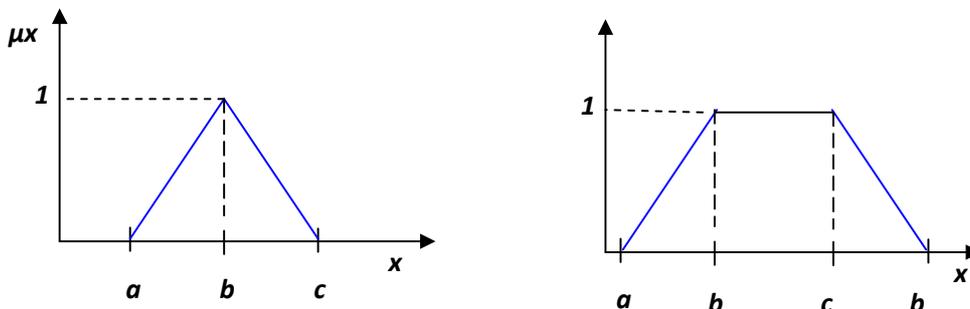


Figure 4.3 *Les fonctions d'appartenance utilisées*

Couche 2 : Chaque nœud dans cette couche correspond à un terme linguistique (excellent, bon, etc.) à une des variables d'entrée dans la couche 1. En d'autres termes. Le lien de la production représente la valeur d'appartenance qui spécifie le degré auquel une valeur d'entrée appartient à un ensemble flou est calculé dans couche 2.

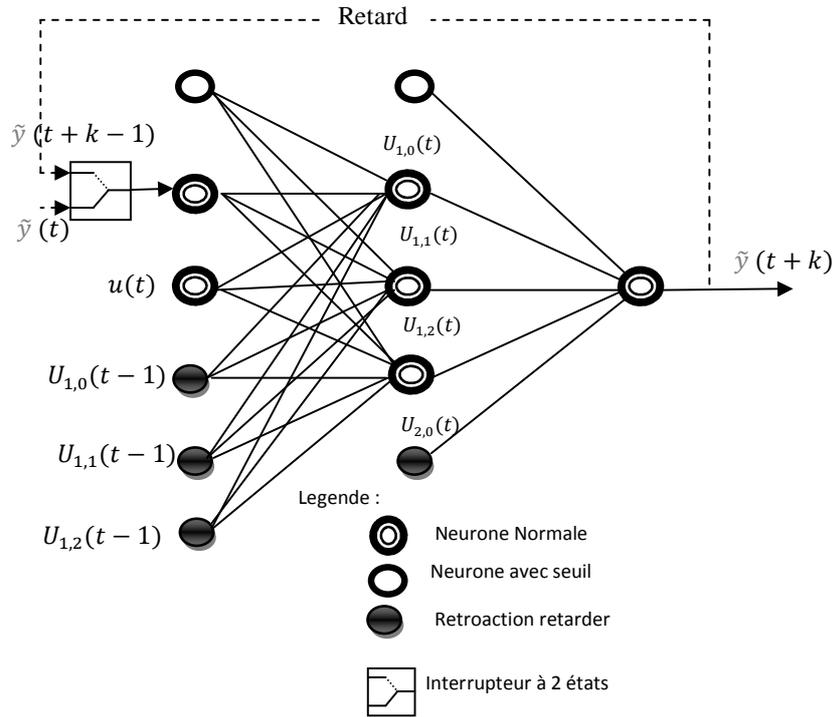


Figure 4.4 La structure de SNFT.

Un algorithme de regroupement détermine le nombre initial et le type de fonctions d'appartenance qui doit être affecté à chacune des variables d'entrées. Les dernières formes des fonctions d'appartenance seront ajustées durant la phase d'apprentissage du système.

Couche 3 : cette couche a plusieurs nœuds, l'un pour former la matrice A et l'autre pour B. Bien que nous pouvons utiliser de nombreux nœuds pour représenter les composants A et B séparément, il est plus commode d'utiliser des matrices.

Ce nœud a deux tâches fondamentalement : combiner les nouveaux antécédents des règles, et déterminer le degré auquel ils appartiennent à la variable linguistique de sorties (haut, moyen, bas, etc.). Le nombre de nœuds dans cette couche sera égal au nombre de règles.

$$\text{for } A \quad o^{[2]} = \sum_{r=1}^R u_r^{[2]} A^r, \text{ for } B \quad o^{[2]} = \sum_{r=1}^R u_r^{[2]} B^r \quad (4.6)$$

Couche 4 : La couche de sortie est constituée d'un unique neurone (dans notre cas). Où n est le nombre de règles floues.

$$o^{[14]} = e^{\frac{(u^{[1]} - c^r)2}{2(s^r)2}} \quad (4.7)$$

Couche 5 (couche contexte) : La couche de contexte est système linéaire à un seul nœud, qui a toutes les sorties des couches 1 et 2 qui lui sont connectés comme entrées. L'utilisation sous forme de matrice d'entrées et de sortie, [voir couche 3]

La couche 5, appelée mémoire statique du réseau, a pour rôle de mémoriser les prototypes des règles, comme pour les réseaux Elman classiques. Les entrées de cette couche ne proviennent plus directement des données d'entrée mais résultent de la sortie de la troisième couche, à savoir la mémoire dynamique. Un premier traitement des données d'entrée est donc effectué par la première couche avant que celles-ci soient mémorisées par la mémoire statique. La règle ainsi mémorisé par les neurones gaussiens aura implicitement pris en compte la dimension temporelle des données d'entrée.

$$o^{[3]} = AX + BU = o_{\text{for A}}^{[2]} o_{\text{context}}^{[1]} + o_{\text{for B}}^{[2]} o_{\text{context}}^{[1]}$$

4.2.1.3 Algorithme d'apprentissage de SNFT

L'apprentissage des systèmes de Neuro-flous temporels est une phase qui permet de déterminer ou de modifier les paramètres du système, afin d'adopter un comportement désiré.

L'étape d'apprentissage est basée sur la descente de gradient d'erreur quadratique moyenne commise par le RNF. Après avoir calculé l'activation en avant des neurones des différentes couches, l'erreur est ensuite rétro propagée dans le sens inverse de l'activation pour pouvoir calculer, pour chaque neurone, sa contribution

$$J_m(U, V) = \sum_{k=1}^N \sum_{i=1}^m (u_{ik})^p (d_{ik})^2 \quad (4.8)$$

Des corrections sont ensuite apportées aux différents paramètres du système (les poids, les seuils et les paramètres des fonctions d'appartenance).

L'utilisateur définit le nombre initial des fonctions d'appartenance pour partitionner les domaines des données d'entrées et spécifie le nombre k, nombre maximum de neurones des règles qui seront créés dans la couche cachée.

Soit l'état initiale $X(0)$ et la sortie désiré $Y_i(t)$, $t=1,2,\dots, te$, l'erreur à l' instant t est définie comme suite :

$$e(t) = \frac{1}{2} \|y_d(t) - y(t)\|^2 = \frac{1}{2} \sum_{i=1}^p [y_{di}(t) - y_i(t)]^2 \quad (4.9)$$

Et l'objectif de l'apprentissage est de minimiser chaque $e(t)$, $t=1,2,\dots$ te.

La technique de descente de gradient est utilisé comme une règle générale : en supposant que w est un paramètre ajustable :

$$\Delta w(t) \propto \frac{\partial e(t)}{\partial w(t)}$$

$$w(t + 1) = w(t) + \Delta w(t) = w(t) - \mu \frac{\partial e(t)}{\partial w(t)} \quad (4.10)$$

Où $\mu > 0$ est le pas d'apprentissage

Les algorithmes d'apprentissage sont détaillés dans l'annexe 2.

4.3 Test de robustesse

Les données Iris de Fisher disponibles ont été proposées en 1933 par le statisticien Ronald Fisher comme données de référence pour l'analyse discriminante et la classification. Celles-ci sont constituées d'un échantillon de 150 fleurs de trois espèces différentes (Iris setosa, Iris virginica et Iris versicolor) et sont décrites par quatre variables mesurant la longueur et l'épaisseur des sépales ainsi que la longueur et l'épaisseur des pétales (dans cet ordre dans le fichier, suivi de la variable espèce). L'objectif de cette section est de mettre en œuvre le réseau de Neuro-Flou temporel d'Elman pour expliquer la classification en ces trois espèces florales grâce aux quatre variables explicatives disponibles (figure 4.5).

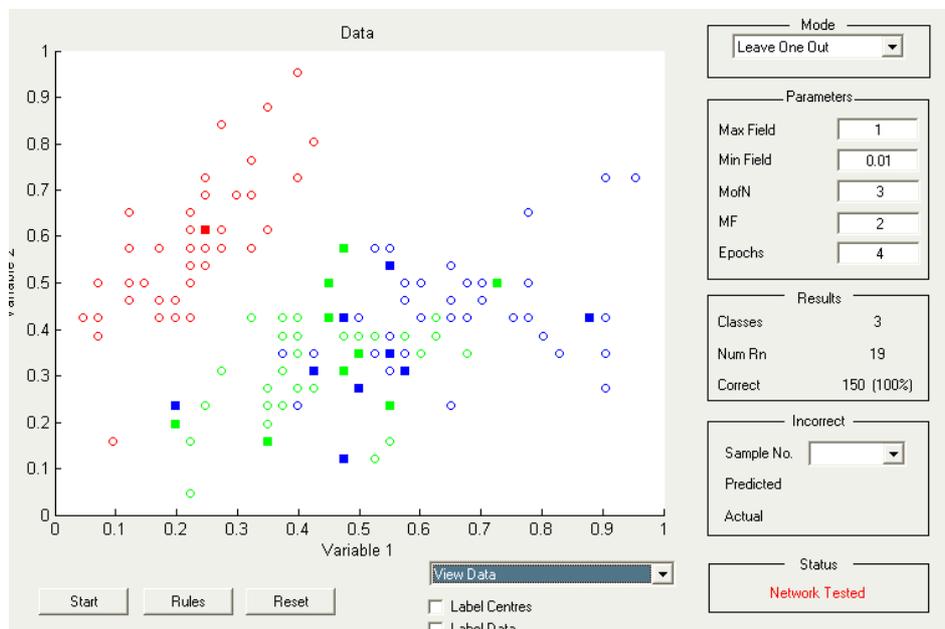


Figure 4.5 *Classification des données Iris*

Codage des sorties : Le codage classique des sorties désirées pour la classification utilise un neurone de sortie par classe, avec une valeur désirée haute pour le neurone de la classe correcte, et une valeur désirée faible pour les autres classes.

Nous allons utiliser le codage suivant :

$$\text{Classe 1} \rightarrow (1\ 0\ 0) \quad \text{classe 2} \rightarrow (0\ 1\ 0) \quad \text{classe 3} \rightarrow (0\ 0\ 1)$$

	Min	max
« longueur sépale »	4	8
« largeur sépale »	2	5
« longueur pétale »	1	7
« largeur pétale »	0	3

Tableau 4.1 *Les rangs de variables d'entrées « Iris ».*

Étape 1 : partition initiale

Cette étape permet de passer du domaine numérique au domaine symbolique, dans lequel on attribue un degré d'appartenance à chaque sous-ensemble flou correspondant aux termes linguistiques associés à la variable.

Ces termes sont choisis en rapport avec le vocabulaire utilisé par l'expert. L'intensité peut prendre le qualificatif « sombre » ou « claire », la surface peut être « petite », « moyenne » ou « grande ».

Le choix du nombre de termes pour qualifier une variable est souvent empirique. Généralement, on retient une équi-répartition des termes et un nombre de termes impair, relativement faible (trois ou cinq) pour ne pas avoir un nombre de règles trop important.

Sur la figure 4.6 les partitions initiales de chaque entrée de donnée par 4 fonctions d'appartenances.

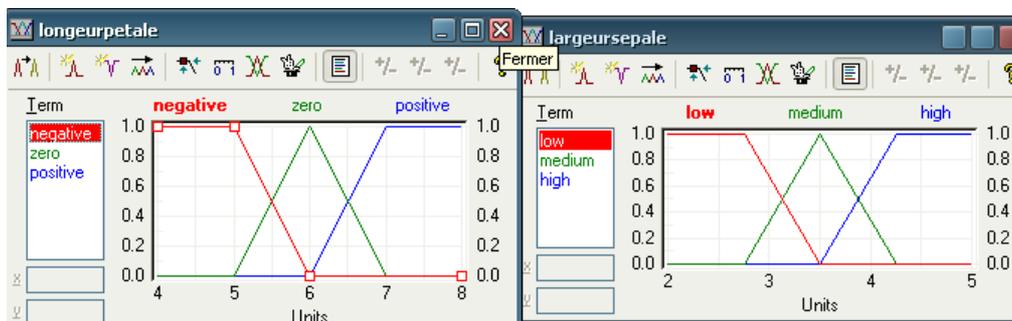


Figure 4.6 *Partition initiale*

Pour l'exemple, on a choisi des fonctions d'appartenances relativement simples. Elles sont linéaires ou en forme de triangle. Mais libre au concepteur du système de choisir une fonction d'appartenance plus complexe s'il estime que le gain de performance est non négligeable. Les fonctions d'appartenances les plus courantes ont une forme de triangle, de trapèze ou de cloche.

Par ailleurs, on remarque que si l'on choisit une fonction d'appartenance en forme de rectangle, ou une fonction de Heaviside, on se retrouve dans le cas d'une logique classique.

Cette première étape de fuzzification va donc traduire les données numériques en différentes variables linguistiques.

Étape 2 : génération des règles floues temporelles

Cette deuxième étape permet la génération de règles floues de type « Si ... alors ... ». Dans notre cas nous quatre paramètres en entrée et trois donnée (classes) de sortie, la forme générale de la règle floue est :

A partir de l'ensemble des points du lot d'apprentissage, un calcul prend en compte l'ensemble des degrés d'appartenance à la classe Ci par rapport à celui des autres classes. En fonction du résultat, on affecte à chaque règle un degré de confiance CF.

Chaque règle délivre une conclusion partielle qui est ensuite agrégée aux autres pour fournir la conclusion finale.

L'inférence repose sur le modèle de Larsen [61] qui utilise comme opérateur de pseudo implication floue la composition max-produit.

Après la phase d'apprentissage nous avons trouvé 5 règles floues qui sont listées sur la figure 4.7

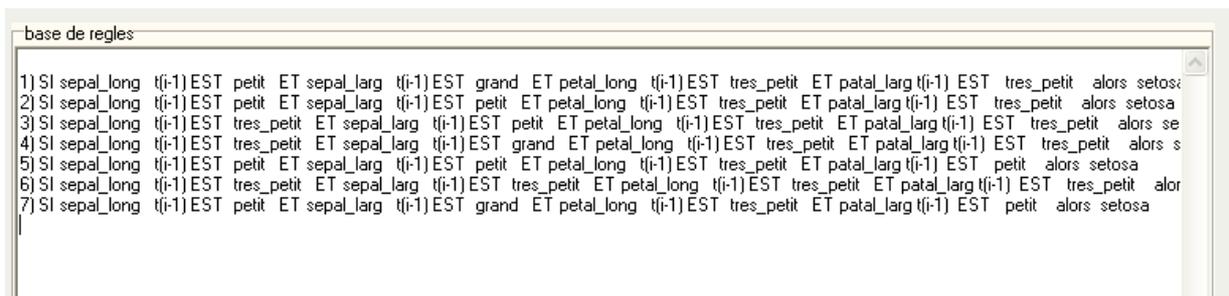


Figure 4.7 La base de règle générée par les données

4.4 Conclusion

Ce quatrième chapitre constitue la première étape de notre contribution. Il a été dédié à la présentation d'un nouveau réseau de Neuro-Flou temporel : système Neuro-Flou Elman avec une technique d'apprentissage basé sur la retro propagation de gradient.

Le SNFT est représenté comme un réseau de neurones flous bouclés pour lequel chaque couche est un composant d'un système Neuro-Flou en utilisant le système d'inférence flou de type 3(TSK). L'apprentissage à deux phases. Un apprentissage de règles, et génère les règles floues temporelles par le parcours de données et optimise les règles par apprentissage des paramètres des ensembles flous qui sont utilisés pour partitionner les données des formes à classer et les paramètres des données.

Dans le chapitre suivant, nous allons évaluer les performances du réseau SNFT avec les règles Floue temporelle , en l'appliquant à des problèmes type de surveillance dynamique. Nous testerons le réseau SNFT sur une problématique d'application des systèmes Neuro-Flous temporels en surveillance.

Pour cela un logiciel appelé NEFSUR (Surveillance Industrielle par les Systèmes Neuro-Flou) est réalisé, ce logiciel est dédié à l'apprentissage, le test, la détection et la classification des défauts et des pannes d'un procédé industriel dédié.

Chapitre V

NEFSUR : une approche Neuro-Floue Proposée

Chapitre V

NEFSUR : une approche Neuro-Floue Proposée

5.1 Introduction

Ce chapitre est réservé exclusivement à la présentation dans un premier temps de notre système d'aide à la surveillance. Il est composé de deux outils: l'un dédié au pronostic et l'autre au diagnostic. Nous présentons notre système d'aide à la surveillance basée sur une architecture Neuro-floue Temporelle, les principes d'apprentissage ainsi que ses liaisons avec l'outil de pronostic.

Dans ce cadre, un logiciel informatique de simulation interactive baptisé NEFSUR (Surveillance par Neuro-Flou Temporel) est développé. Ce logiciel consacré essentiellement à la création, l'apprentissage et au test d'un système Neuro-flou de classification des pannes d'un procédé industriel. NEFSUR est composé de deux sous système NEFPROG (Neuro-Fuzzy Prognosis) et NEFDIAG (Neuro-Fuzzy Diagnosis).

A ce jour, aucun outil de RdF n'est appliqué à la cimenterie d'Ain Touta-Batna. Néanmoins durant les années (1987-1997), la logique floue est intervenue à la conduite automatique de la zone four. Pour cela, nous avons développé un nouveau système dans le domaine de la RdF, utilisant la logique floue et les RNAs pour le diagnostic prédictif pour la zone de cuisson de la cimenterie, notre champ d'application.

5.2 Conception d'un outil Neuro-Flou Temporel pour la surveillance dynamique

Dans le contexte de la surveillance dynamique, une forme est l'observation simplifiée de l'état du système. La classification d'une forme est réalisée par un module de classification, appelé classifieur, en utilisant une règle de décision. Le problème de la RdF est donc de caractériser le modèle et l'étiquette de chaque classe associée à une forme. Cela requiert le recours à des techniques de classification permettant de regrouper, par apprentissage, les formes similaires. Le diagnostic par RdF devient alors un problème de classification. Le mode de fonctionnement actuel du système est déterminé en connaissant la classe de l'observation actuelle de l'état de fonctionnement du système. Le modèle de chaque classe est représenté par une fonction d'appartenance qui détermine la valeur d'appartenance d'une forme à chaque classe.

La surveillance en ligne d'un système industriel a pour objectifs de détecter tout dysfonctionnement du procédé, de localiser le ou les éléments défaillants et de déterminer la ou les causes de défaillances. Ces informations sur l'état du système peuvent alors être transmises aux opérateurs qui peuvent agir en conséquence et/ou être utilisées pour modifier, reconfigurer la commande voire le système lui-même ou sa mission. Dans cette section, nous présentons les étapes de la surveillance dynamique par la méthode numérique ou/et symbolique: les systèmes Neuro-Flous Temporels.

Quand le diagnostic est basé sur des observations multiples, elles sont regroupées pour former des classes qui définissent une situation ou un mode de fonctionnement du processus,

Au quel une nouvelle observation sera comparée pour être identifiée. En d'autres termes, le diagnostic a pour mission d'identifier le mode de fonctionnement d'un système à partir d'observations sur celui-ci (figure 5.1). [77] [80].

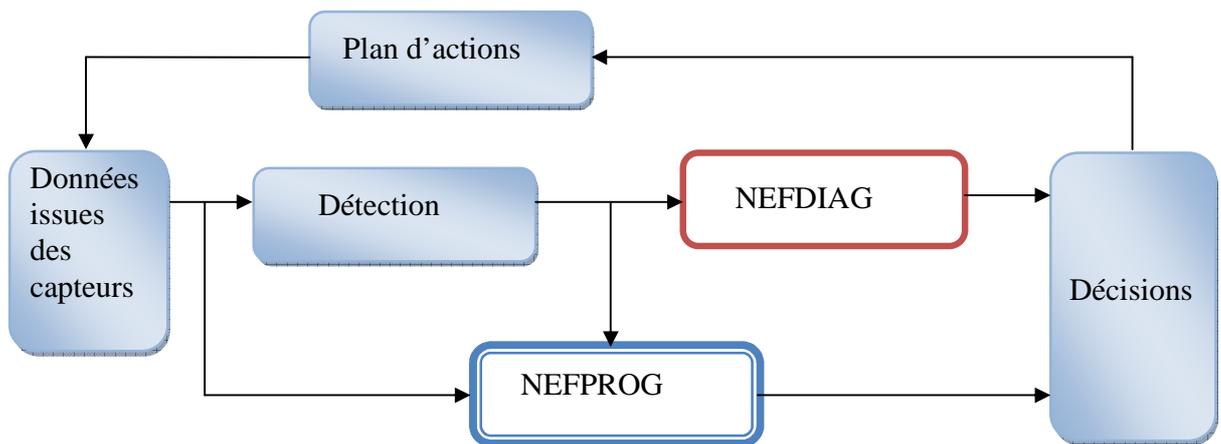


Figure 5.1 Positionnement de NEFPROG et NEFDIAG DANS NEFSUR

La méthode proposée concerne toutes les phases de la fonction de surveillance: la détection et le diagnostic de défaillance.

- *L'outil de détection dynamique.* En entrée de l'outil de détection, nous trouvons les informations données par les capteurs. Ces données sont traitées dynamiquement. Les sorties donnent les modes opératoires (symptômes) de l'équipement surveillé. Nous utilisons les réseaux de neurones flous récurrents.
- *L'outil de pronostic et diagnostic.* L'entrée de l'outil de diagnostic reprend les sorties du système de détection : les modes opératoires. Nous trouvons aussi en entrée des données de type qualitative ou quantitative que l'opérateur pourra ajouter pour améliorer le diagnostic. En sortie, nous trouverons les différentes causes possibles associées à un degré de crédibilité et un degré de sévérité pour chacune d'elles. Ces degrés aideront le responsable

maintenant à évaluer et planifier les actions de maintenance. En suivant les conclusions des chapitres précédents nous utilisons les réseaux de neurones ainsi que la logique floue dans un seul et même outil appelé réseau Neuro-Fou Temporel (figure 5.2).

Le fonctionnement d'un système de diagnostic par RdF se déroule en trois phases : une phase d'analyse, une phase de choix d'une méthode de décision floue (inférence floue) et une phase d'exploitation [71].

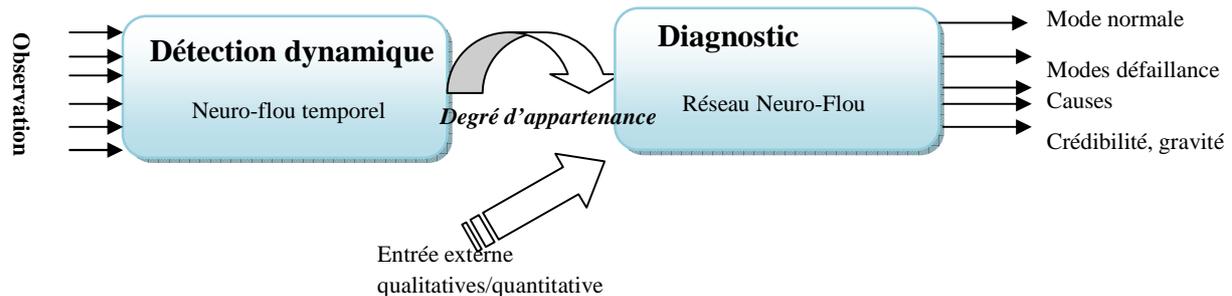


Figure 5.2 Schéma du système d'aide à la surveillance

Le diagnostic de défaillances dans ce cadre est essentiellement vu comme un problème de classification. Le but principal est de construire un bloc de correspondance tel qu'à partir d'un ensemble d'informations décrivant la situation courante de processus, il est possible d'obtenir les causes probables de situations anormales [74] [75].

5.2.1 Représentation de NEFDIAG

Un logiciel informatique de simulation interactive baptisé NEFDIAG (Neuro Fuzzy DIAGnosis) version 0.0 est développé au sein de LAP (Batna.). Ce logiciel est consacré essentiellement au développement, à l'apprentissage et au test d'un système Neuro-Flou de classification des pannes d'un procédé industriel.

NEFDIAG peut être représenté comme un type spécial de perceptron flou, à trois couches utilisé pour classifier des défaillances.

NEFDIAG fait son apprentissage par un ensemble de formes, telle que chaque forme soit affectée (classée) vers une des classes prédéfinies. NEFDIAG génère les règles floues par un parcours de données et optimise ensuite les règles par apprentissage des paramètres des sous ensembles flous qui sont utilisés pour partitionner les données « caractéristiques » des formes à classer et les paramètres des données. NEFDIAG représente un classificateur flou avec un ensemble de classes $C=\{c1,c2,\dots,c_m\}$.

Après l'apprentissage, NEFDIAG peut être utilisé pour classifier une nouvelle observation, le système peut être représenté sous forme de règles floues :

Si	symptôme1	est	A1
	symptôme2	est	A2
	symptôme3	est	A3
	symptômen	est	An

Alors la forme $(x_1, x_2, x_3, \dots, x_n)$ est à la classe mode de défaillance 1.

Telque A1, A2, A3, An sont des termes linguistiques représentés par des ensembles flous. Cette caractéristique nous permet de connaître des analyses sur nos données, et utilise ces connaissances pour les classifier.

La tâche du système NEFDIAG est de découvrir « créer les règles par les ensemble de formes d'apprentissage » les règles et l'apprentissage des fonctions d'appartenance pour déterminer la mode de défaillance exact pour une observation quelconque.

- **initialisation** : pour chaque donnée issue des capteurs, il y a une unité d'entrée, et pour chaque mode de défaillance il y a une unité de sortie. Pour chaque unité d'entrée une partition floue initiale est spécifiée (exemple, nombre des fonctions d'appartenance triangulaire).
- **Apprentissage des règles** : Le système NEFDIAG peut démarrer avec une base de connaissance partielle des formes, et raffiner durant l'apprentissage. La règle sera créée par la recherche (pour une forme donnée f) la combinaison des fonctions d'appartenance telle que chaque entrée produit la plus grande fonction d'appartenance. Si cette combinaison n'est pas identique pour les règles existantes dans la base des règles et le nombre de règles n'est pas maximum, alors une règle sera créée et ajoutée à la base des règles.
- **Apprentissage des FA** : pour l'apprentissage des fonctions d'appartenance, une simple rétro propagation sera utilisée. Une décision sera créée qui dépend de l'erreur de sortie pour chaque unité de règles. Chaque règle change ses fonctions d'appartenance par le changement de leurs supports.

Les algorithmes présentés dans cette section sont des extensions d'une approche proposée par [53]. Ils sont utilisés dans des logiciels « NEFCLASS, NEFCON, NEFPROX ».

L'algorithme d'apprentissage des règles de l'approche proposée utilise les poids des règles et détermine les meilleures règles par la « mesure de la performance » ou par une valeur qui indique l'efficacité d'une règle quelconque. En plus l'algorithme essaie de faire une réduction de la taille de la base de règles, par la sélection des règles en utilisant leurs performances en classification des modes de défaillances, ou par le déroulement des données d'apprentissage.

*Algorithme: Apprentissage des règles floues*début

- (1) sélectionner la forme suivante (f,t) de Γ .
- (2) pour chaque unité d'entrée $x_i \in U_1$ chercher la fonction d'appartenance $\mu_{ii}^{(i)}$

$$\mu_{ji}^{(i)}(p_i) = \max_{j \in \{1, \dots, q_i\}} \{\mu_j^{(i)}(x_i)\}$$

- (3) si le nombre des règles $\leq k_{max}$, et il n'existe pas de règle R avec :

$$W(x_1, r) = \mu_{i1}^{(1)}, \dots, W(x_n, R) = \mu_{in}^{(n)}$$

Alors créer un nœud et connecter la au nœud de sortie CI si $t_i=1$.

- (4) s'il y a des formes non traitées dans Γ et $k \leq k_{max}$ alors aller a (1). Sinon

Fin.**Algorithme 5.1.** *Apprentissage des règles floues.*

L'algorithme d'apprentissage détecte (calcule) tous les antécédents des règles et crée ensuite la liste des antécédents. Dans un premier temps, cette liste est vide, ou contient des antécédents de règles de connaissances à priori.

L'algorithme sélectionne un conséquent pour chercher l'antécédent A et crée la liste de base de règles candidates.

Pour chaque antécédent, une mesure de performances des règles $p \in [-1, 1]$ est calculée ;

Pour $p=1$, la règle est générale et classe correctement.

Pour $p=-1$, la règle classe toutes les formes incorrectement.

Pour $P=0$ quelques formes sont classées correctement et autres non.

Seulement les règles avec $p > 0$ sont considérées pour l'utilisation durant toute la deuxième phase d'apprentissage.

Algorithme : Apprentissage des ensembles flous.

Début

1- sélectionner la forme suivante (p,t) de Γ , propager la, par NEFDIAG et détermine le vecteur de sortie C .

2- pour chaque unité de sortie C_i ; déterminer la valeur de delta

$$\delta C_i = t_i - O_{C_i}$$

3- pour chaque unité de règle R avec $O_R > 0$

a- déterminer la valeur de δ

$$\delta_R = O_R(1 - O_R) \sum_{C \in U_3} W(R, C) \delta_C .$$

b- trouver x' tel que : $w(x', R)(o_{x'}) = \min_{x \in U_1} \{w(x, R)(\delta_C)\}$

c -pour l'ensemble flou $w(x', R)$ déterminer les valeurs de δ pour les paramètres a, b, c utilisant le pas d'apprentissage $\sigma > 0$.

$$\delta_b = \sigma \cdot \delta_R \cdot (c-a) \cdot \text{sgn}(O_{x'} - b),$$

$$\delta a = -\sigma \delta_R \cdot (c-a) + \delta_b,$$

$$\delta c = \sigma \delta_R \cdot (c-a) + \delta_b.$$

Appliquer le changement aux $w(x', R)$.

4 -si l'itération est terminée, ou les critères de fin sont validé ; **alors stop**

Sinon aller a (1).

Fin

Algorithme 5.2. Apprentissage de fonctions d'appartenance

5.3 Modélisation et simulation du procédé de fabrication du Clinker

5.3.1 Brève présentation de l'entreprise

Nous allons exploiter la caractéristique au cours de développement de notre application sur le processus de clinkerisation objet de cette section.

Cette cimenterie d'une capacité de 2.500.000 t/an (2 fours) est composée de plusieurs unités qui déterminent les différentes phases du processus de fabrication du ciment. La carrière d'où sont extraites les matières premières (argile et calcaire), les stations de concassage destinées à réduire la taille des blocs de matières (calcaire et argile 1 000 t/h, les ajouts 150 t/h).l'atelier de broyage de cru est composé de deux broyeurs de 140t/h chacun. L'atelier de cuisson regroupe deux fours dont le débit clinker est de 1560 t/h. le broyage de ciment comprend deux broyeurs de 100t/h chacun. Les expéditions du ciment s'effectuent à partir de deux stations, une pour les camions et une pour les wagons (figure 5.3).

La description complète des différents aspects des procédés de transformation nécessite différents types d'informations représentant les éléments de base pour la modélisation et la simulation du procédé [77].

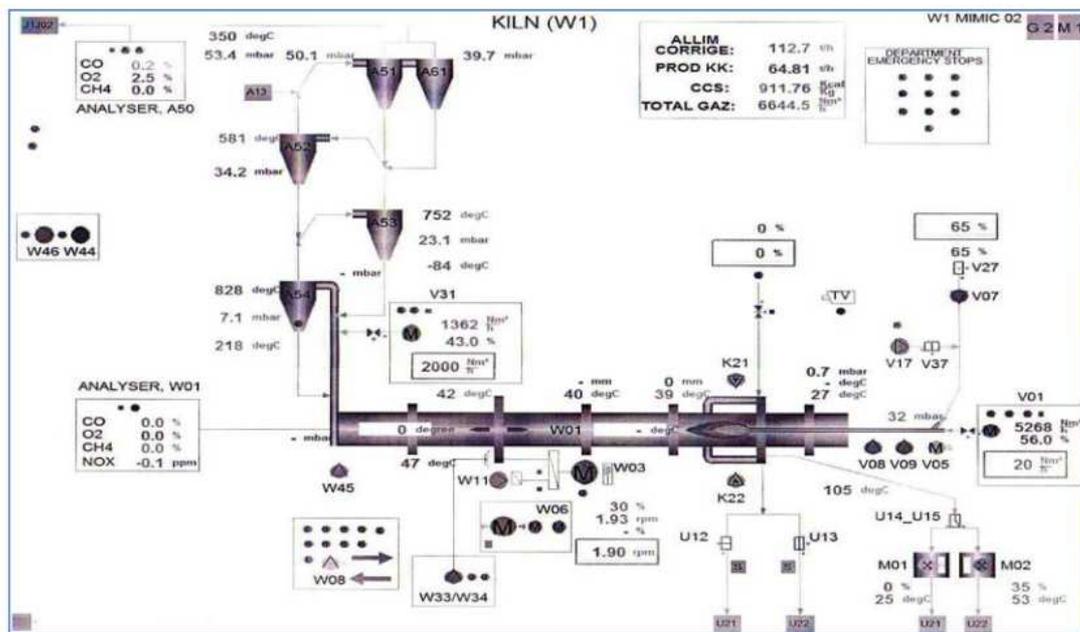


Figure 5.3 Schéma synoptique de l'atelier de clinkerisation

5.3.2 Simulation du fonctionnement normal

La simulation du processus de clinkerisation en fonctionnement normal a permis de représenter l'atelier de clinkerisation en montrant les étapes du passage de la charge et des gaz chauds dans les différents échangeurs (figures 5.4)

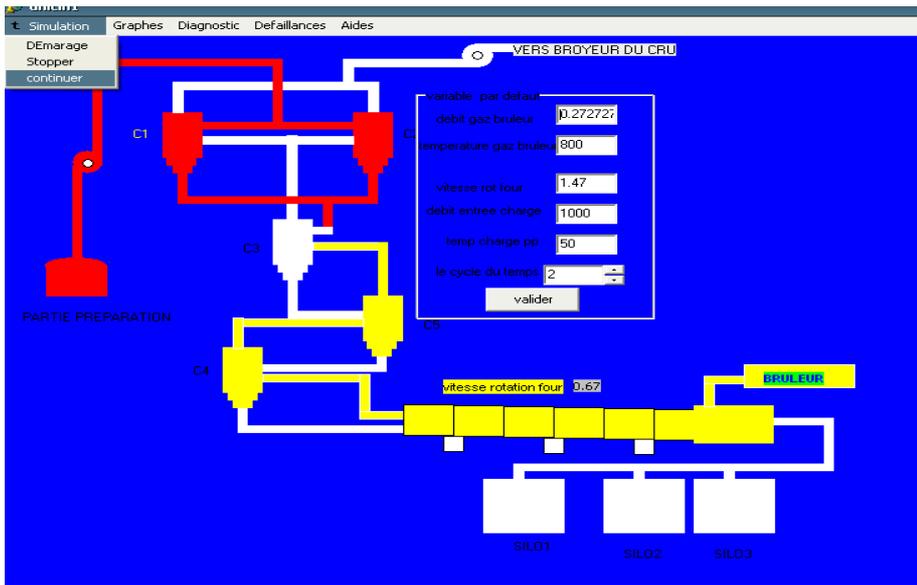


Figure 5.4 Représentation de l'atelier de clinkérisation après introduction des gaz chauds et début l'arrivée de la charge

Les régulations sont faites au niveau des trémies en vérifiant le niveau de chaque trémie, et selon la vitesse des tabliers et les doseurs soient en marche ou en panne, et selon le point de consigne donné par l'opérateur (figure 5.5). La quantité extraite de chaque trémie est calculée selon le pourcentage désiré pour le mélange du broyeur ciment.

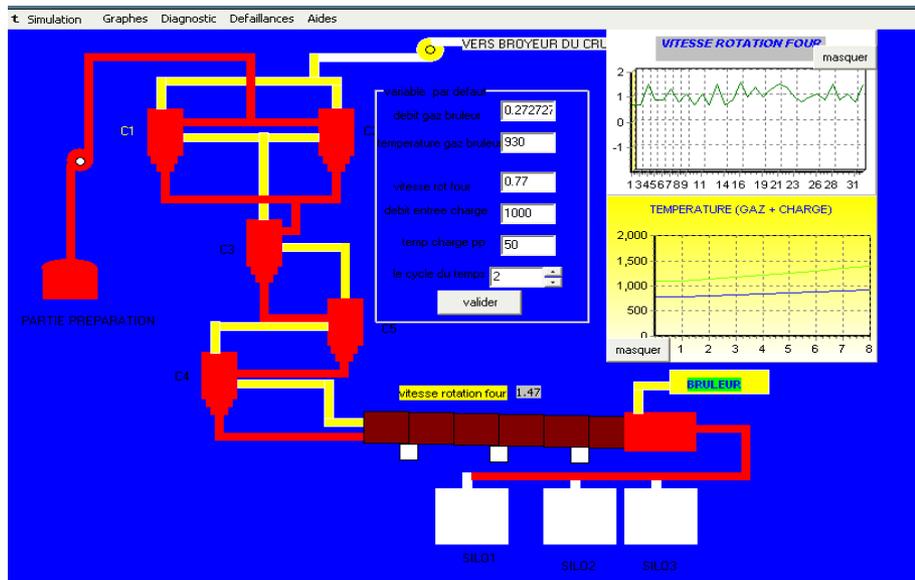


Figure 5.5 Représentation de l'atelier de clinkérisation après passage de la charge dans la tour de préchauffage

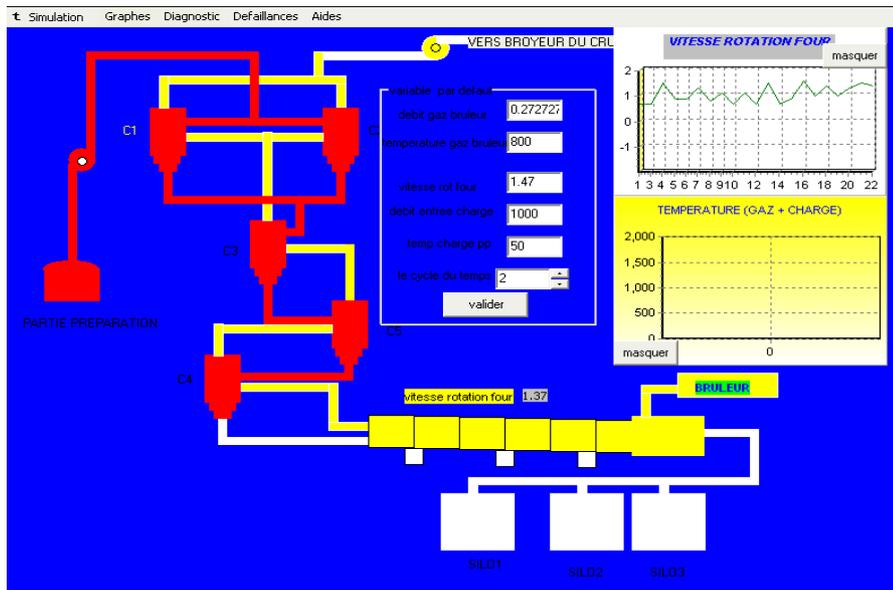


Figure 5.6 Représentation de l’atelier de clinkérisation après passage de la charge dans tout l’atelier

Par ailleurs, et pour meilleure interactivité avec l’utilisateur de cette plateforme, l’évolution des variables significatives est représentée par des tables et des graphes qui signalent les valeurs prises par ces variables à des intervalles bien déterminés (figure 5.6) .

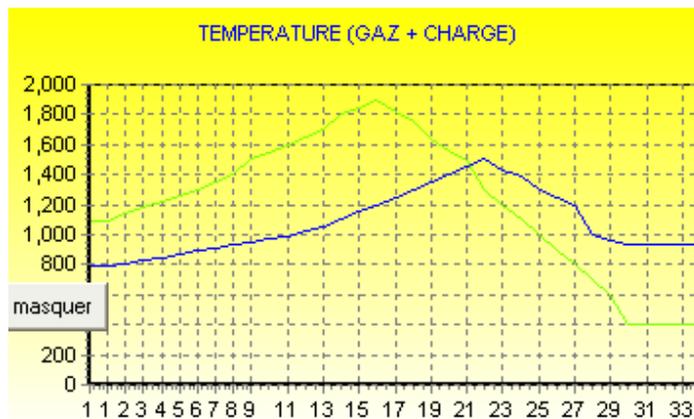


Figure 5.7 Le profil thermique la longueur du four

5.4 Pronostic de défaillance

Afin de tester l’influence de différentes architectures du système Neuro-Flou Temporel sur les performances de prédiction d’un système réel, nous avons réalisé le logiciel NEFSUR.

Nous allons tester le système Neuro-Flou Temporel sur une application où la dynamique des données d’entrée est tout aussi importante. Cette application concerne la prédiction de séries temporelles pour des applications de pronostic industriel. Nous allons tester les propriétés dynamiques du système Neuro-Flou Temporel sur des données temporelles.

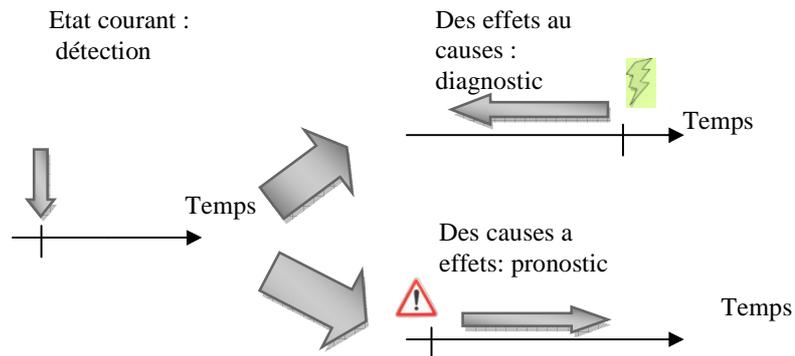


Figure 5.8 *Detection, diagnostic and prognostic: l’ aspect phenomenologique*

Une application de prédiction temporelle est complètement différente de la problématique de la reconnaissance de séquences temporelles. La prédiction temporelle est considérée comme une application d’approximation de fonctions et non de classification. Les variables de sortie du système Neuro-Flou sont donc des variables de type réel et non catégoriel et la couche de sortie du réseau est constituée par des neurones linéaires.

Les variables de notre système Neuro-flou utilisées sont présentées sur le Tableau 5.1. Donc, nous avons un système Neuro-Flou de 20 entrées et 4 sorties qui ont été créées pour faire un pronostic du système. Les règles créées avec le système sont des connaissances a priori, c.à.d. La base de règle à priori. Chaque variable à une partition initiale est modifiée au long de la phase d’apprentissage (nombre de ensembles flous pour chaque variable).

	Paramètres	Description des paramètres	Point de consigne état normale
P1	W01S1	Vitesse four.	$1 < W01S1 < 3t/mm$
P2	A50T1	Température gaz sortie cyclone A50.	$300 < A50T1 < 350^{\circ}C$
P3	A52T1	Température gaz sortie cyclone A52.	$530 < A52T1 < 555^{\circ}C$
P4	A52P2	Pression cyclone A52.	$25 < A52P2 < 35mbar$
P5	A53T1	Température gaz cyclone A53	$700 < A53T1 < 760^{\circ}C$
P6	A53T2	Température matière cyclone A53	$700 < A53T1 < 760^{\circ}C$
P7	A53P1	Pression cyclone A53.	$18 < A53P1 < 20mbar$
P8	A54T1	Température gaz cyclone A54	$820 < A54T1 < 860^{\circ}C$
P9	A54P2	Pression cyclone A54t.	$8 < A54P2 < 10mbar$
P10	COP	Teneur CO boîte à fumée.	$0 < COP < 0.25\%$
P11	O2C	Teneur O2 sortie cyclone A50	$0.1 < O2C < 0.6\%$
P12	V31F1	Début gaz boîte à fumée.	$850 < V31F1 < 893Nm^3/h$
P13	V01F1	Début gaz four.	$3280 < V01F1 < 5056 Nm^3/h$
P14	W01X1	Moment four	$20 < W01X1 < 60\%$
P15	COC	Teneur CO sortie cyclone A50	$0 < COC < 0.25\%$
P16	TV	Température virole four	$1000 < TV < 1800^{\circ}C$
P17	V07P1	Pression air primaire.	$28 < V07P1 < 30mbar$
P18	U01T1	Température clinker.	$100 < U01T1 < 150^{\circ}C$
P19	A54T2	Température matière cyclone A54	$820 < A54T2 < 830^{\circ}C$
P20	K01T1	Température air secondaire.	$500 < K01T1 < 900^{\circ}C$

Tableau 5.1 Description des paramètres de vecteur d'état [95]

5.4.1 Modèles de simulation

Nous avons réalisé des simulations sur une architecture de prédiction correspond au system unique Neuro-Flou Temporel à trois entrée (figure 5.9), tel que :

$$\{U(t-b), u(t-c), y(t-a)\}, a=1\dots 4 \text{ et } b, c=1..6$$



Figure 5.9 Le modèle de simulation

La prédiction est considérée comme processus basé sur l'agrégation des états actuels du système, mais également, du futur. En effet, nous avons considéré qu'il est utile de tenir compte des futures actions, telle que la modification du profil de mission due à une intervention externe ou à une action programmée de maintenance (figure 5.10).

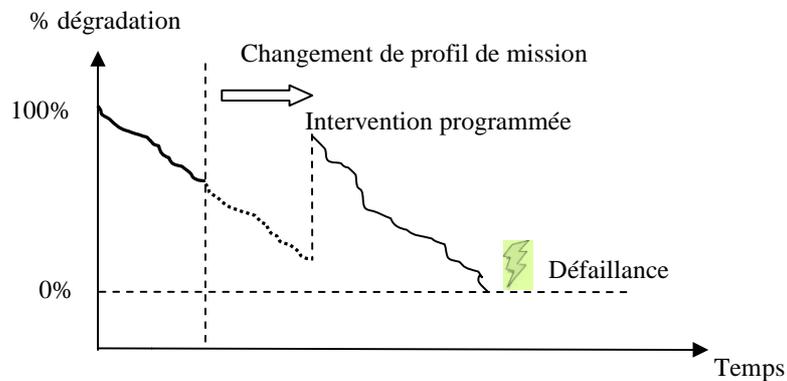


Figure 5.10 Le changement de profil de mission

En conséquence, le juste-dans-temps-point (la période de la défaillance quand la durée de la vie est [0%]) obtient une autre dimension liée au point de départ ([100%] de la durée de la vie de machine) [14].

Ainsi, la première structure NEFPROG a pour entrées $\{u(t-b), y(t-a)\}$, et la deuxième sortie $y(t+prev/2)$ du premier ANFIS et les sollicitations et interventions programmées sous la forme de $u(t+prev/2-c)$ [14].

Pour juger des résultats et conformément à la section 2.3, nous avons utilisé:

- d'abord, le RMSE, qui est le plus commun des indicateurs d'erreur,
- ensuite, le MAPE qui permet de s'affranchir de l'ordre de grandeur des signaux prédits.

Ces indicateurs agrégés ne permettent pas systématiquement de bien évaluer la dispersion des erreurs de prédiction et doivent être complétés par l'analyse de la fonction de densité de probabilité des erreurs sur chacun des tests.

Notre modèle correspond à un unique système Neuro-Flou Temporel avec trois entrées.

L'erreur en valeur absolue varie de ($\min_e = -0.0664$, $\max_e = 0.0478$) pour $\text{prev}=1$ à ($\min_e = -0.6119$, $\max_e = 0.1574$) pour $\text{prev}=20$.

L'architecture donne des résultats satisfaisants pour des prévisions à court terme et même à moyen terme. Soit la série temporelle de vitesse de rotation du four avec une taille de 1200 échantillon (figure 5.11) ; Les données que nous allons utiliser dans cette partie ont été obtenues avec les conditions suivantes : $\Delta = 17$, $x(0) = 1.2$ $x(t - \Delta) = 0$ et pour $0 \leq t < \Delta$ avec une fréquence d'échantillonnage de 1. La base de données contient 1200 points : les 500 premiers points à partir du 118^{ème} point sont utilisés pour la phase d'apprentissage, le reste des points (à partir du 618^{ème} point) pour le test du réseau. L'apprentissage ainsi que le test de prédiction du réseau de neurones se fera sur un horizon de six pas : $x(t+6)$

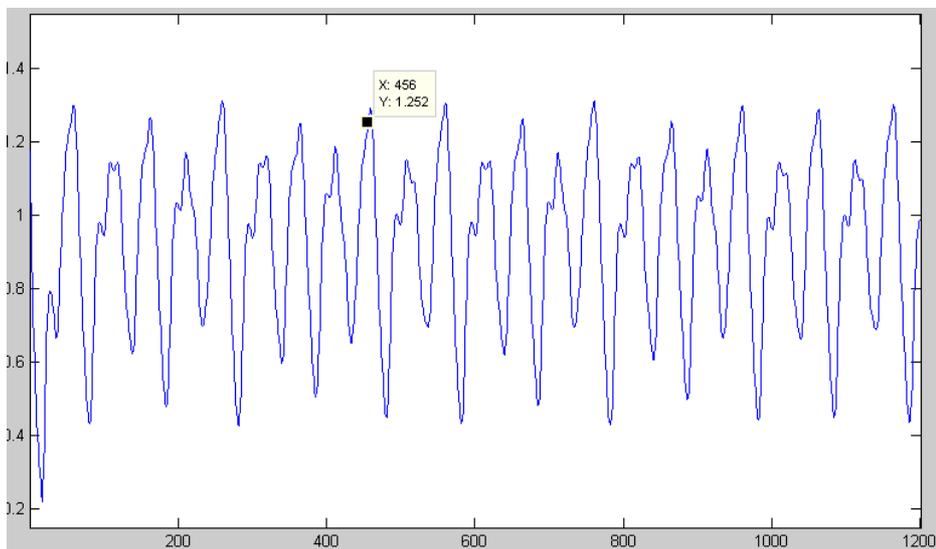


Figure 5.11 Série temporelle vitesse rotation four

Après avoir donné la structure du système NFTE (figure 5.12), nous avons besoin d'associer à chaque variable d'entrée un nombre fini d'ensembles flous et utiliser le type de fonction d'appartenance associée, et aussi l'intervalle de définition des variables et ou ensembles flous. Toutes ces configurations, ont l'avantage d'accélérer la vitesse d'apprentissage et aussi l'orientation du système Neuro-Flou pour une meilleure exploitation des ressources.

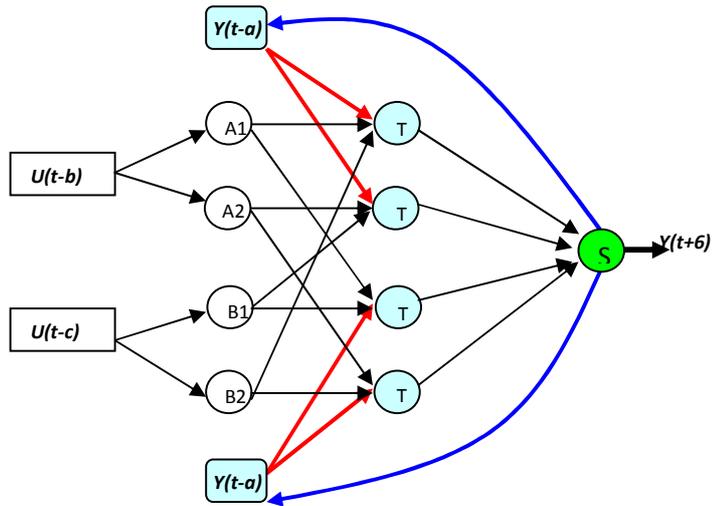


Figure 5.12 LE SNFT pour la prédiction de $y(t+6)$

Etape.1 : après détermination de toutes les variables d’entrée et les paramètres de notre système, NEFSUR donne d’abord les partitions de chaque variable d’entrée. La figure 5.13 représente les paramètres et les conditions d’apprentissage et de test du NEFSUR, La figure 5.14 représente la partition de variable d’entrée « vitesse de rotation du four » avant et après la phase d’apprentissage.

Une entrée est caractérisée par son domaine, et sa partition floue, c'est-à-dire les sous-ensembles flous qui la composent. Elle peut être active (par défaut) ou inactive. Le domaine de variation d'une entrée est par défaut (selon le cahier de charge) [0.2--1.8].

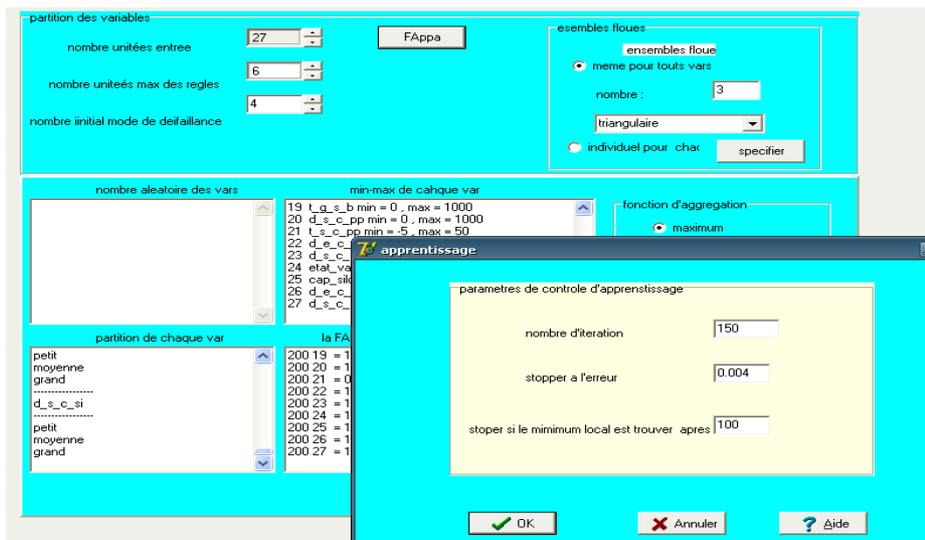


Figure 5.13 Les conditions d’apprentissage et de test

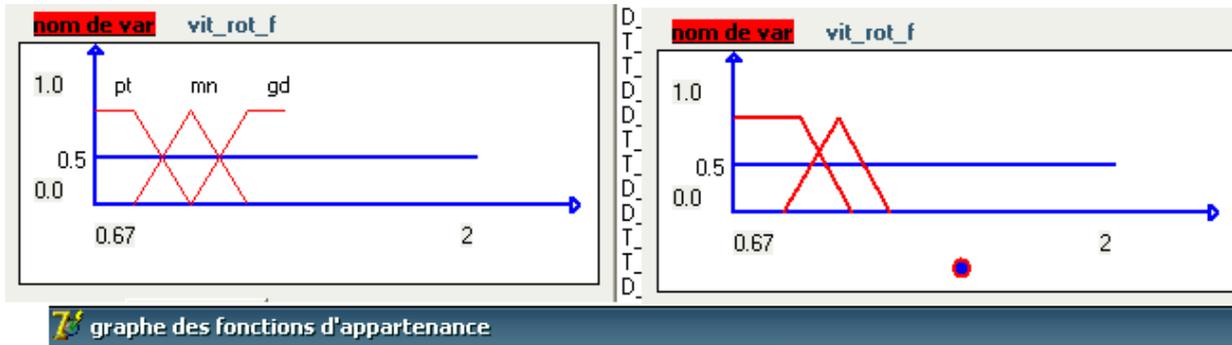


Figure 5.14 Les fonctions d'appartenance avant apprentissage

Figure 5.15 Les fonctions d'appartenance après apprentissage

Etape.2 : génération des règles floues temporelles :

La simplification peut s'appliquer à tout SIF, indépendamment de la méthode de génération utilisée pour le construire. Elle ne modifie pas les SEF des partitions, mais seulement les règles.

L'optimisation peut s'appliquer à tout SIF, indépendamment de la méthode de génération utilisée pour le construire. Elle peut modifier tous ses éléments, y compris les SEF des partitions.

La figure 5.16 représente des règles floues temporelles pour la prédiction de la variable vitesse de rotation du four et l'exploiter pour un pronostic moyenne terme.

La seconde étape est celle de la sélection des règles (phase 1), puis de l'optimisation de leurs conclusions (phase 2). L'algorithme travaille en fonction d'une sortie (Défaut : la première sortie, ou la dernière colonne du fichier de données (figure 5.18) [82].

```
base de regles
1) Si vit_rot_f (t) EST grand vit_rot_f (t-a) est grand et vit_rot_f (t-b) et Alors vit_rot_f (t) est grand
2) Si vit_rot_f (t) EST petit vit_rot_f (t-a) est grand et vit_rot_f (t-b) et Alors vit_rot_f (t) est grand
3) Si vit_rot_f (t) EST petit vit_rot_f (t-a) est petit et vit_rot_f (t-b) et Alors vit_rot_f (t) est moyen
4) Si vit_rot_f (t) EST grand vit_rot_f (t-a) est petit et vit_rot_f (t-b) et Alors vit_rot_f (t) est petit
5) Si vit_rot_f (t) EST petit vit_rot_f (t-a) est moyen et vit_rot_f (t-b) et Alors vit_rot_f (t) est petit
6) Si vit_rot_f (t) EST moyen vit_rot_f (t-a) est grand et vit_rot_f (t-b) et Alors vit_rot_f (t) est moyen
```

Figure 5.16 La base de regèle associées

La figure 5.17 nous montre qu'il existe une corrélation linéaire entre la série prédite et la série observée. Ce coefficient de corrélation linéaire est calculé et est égal à $R^2 = 0,90$.

Afin d'apprécier le modèle de prédiction de la moyenne de vitesse de rotation du four réalisé, nous avons calculé l'erreur de prédiction pour chaque vitesse de rotation du four

prédite. L’histogramme de l’erreur de prédiction de la vitesse de rotation du four (figure 5.18) nous montre que l’erreur varie entre - 2 m/s et + 2 m/s.

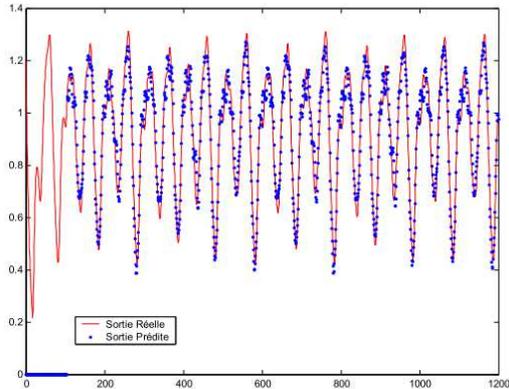


Figure 5.17 Comparaison de la sortie réelle avec celle donnée par le SNFT [80]

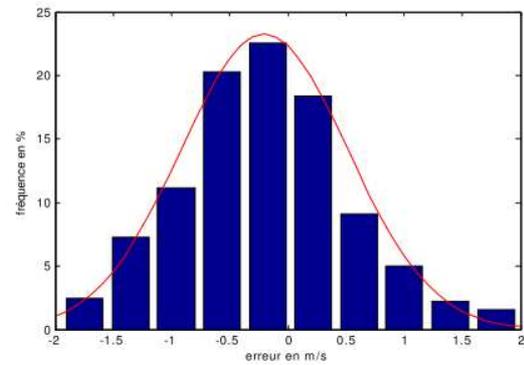


Figure 5.18 Histogramme de l’erreur de prédiction de la vitesse rotation four [81]

D’après l’histogramme de la figure 5.19, nous pouvons noter que, environ 22% des données de validation présente une erreur de prédiction presque nulle, environ 60% des données de validation présente une erreur absolue de prédiction inférieure ou égale à 0,5 m/s et environ 85%

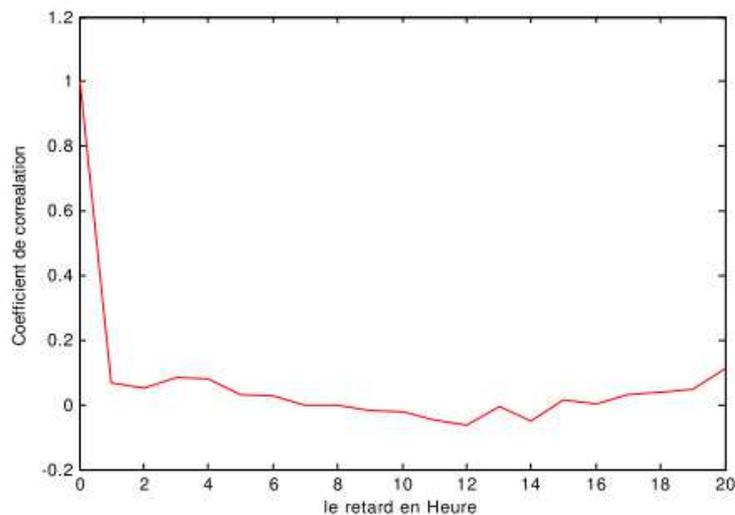


Figure 5.19 Autocorrelogramme de l’erreur de prédiction de la vitesse du four

Des données de validation présente une erreur absolue de prédiction inférieure ou égale à 1m/s. L'erreur de prédiction peut être modélisée sous forme d'un bruit blanc gaussien de moyenne égale à -0,2005 m/s et d'écart type égale à 0,7207 m/s. Ceci est vrai car l'autocorrélogramme (figure 5.20) nous montre que l'erreur de prédiction est non corrélée et que la moyenne et l'écart type de l'erreur de prédiction ont respectivement pour valeur - 0,2005 m/s et 0,7207 m/s.

5.5 Phase de détection dynamique

Pour détecter les défaillances du système, il faut être capable de classer les situations observables comme étant normales ou anormales [78] [79].

Cette classification n'est pas triviale, étant donné le manque d'information qui caractérise généralement les situations anormales. Une simplification communément adoptée consiste à considérer comme anormale toute situation qui n'est pas normale.

Nous nous intéressons au problème d'unité de clinker, en basant notre étude sur l'AMDEC effectuée par M.D Mouss [16], Cette étude propose 5 classes de sorties avec les répartitions suivantes :

1. Décarbonatation perturbée (mauvaise)	Mode de défaillance (MD1)
2. Mauvaise clinkerisation	Mode de défaillance (MD2)
3. mauvais refroidissement	Mode de défaillance (MD3)
4. Insuffisance ou absence du gaz	Mode de défaillance (MD4)

Tableau 5.2 modes de défaillance [75]

5.5.1 Modèles de simulation de la détection

Après avoir donné la structure du système Neuro-Flou Temporel, nous avons besoin d'associer à chaque variable d'entrée un nombre fini d'ensembles flous et utiliser le type de fonction d'appartenance associée, et aussi l'intervalle de définition des variables et ou ensembles flous (figure 5.20). Toutes ces configurations, ont l'avantage d'accélérer la vitesse d'apprentissage et aussi l'orientation du système Neuro-Flou pour une meilleure exploitation des ressources [73] [74].

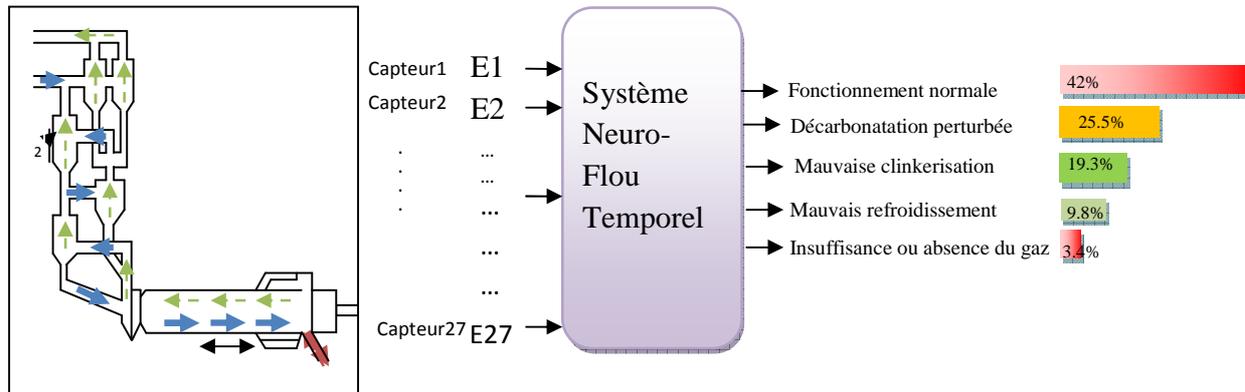


Figure 5.20 Le Modèle de simulation pour la détection

Suite à une étude présentée dans [84] [85] [86], nous utilisons pour la phase de détection le NEFPROG. Son fonctionnement est dynamique et ses entrées sont reliées aux différents capteurs du système à surveiller. En sortie, nous trouvons les modes opératoires de ce système. Son apprentissage s’effectue en deux étapes « génération des règles floues et optimisation des ensembles flous » (Annexe 2)

Après détermination de toutes les variables d’entrée et les paramètres de notre système, NEFSUR donne d’abord les partitions de chaque variable d’entrée.

La figure 5.21 représente les paramètres et les conditions d’Apprentissage et de test du SNFT, la partition de variable d’entrée « débit entrée charge four» avant et après la phase d’apprentissage.

Le problème d’apprentissage (création et raffinement) des règles floues consiste à combiner toutes les possibilités des ensembles flous. Supprimer toutes règles n’a aucune importance pour la reconnaissance. Ensuite le système va faire le deuxième passage pour régler les paramètres des ensembles flous (figure 5.22).

Le système NEFSUR peut démarrer avec une base de connaissances partielle des formes, et raffiner durant l’apprentissage. L’avantage principal de l’utilisation de la base de règles floues réside dans sa modularité et sa facilité d’extension (suppression ou rajout d’autres règles).

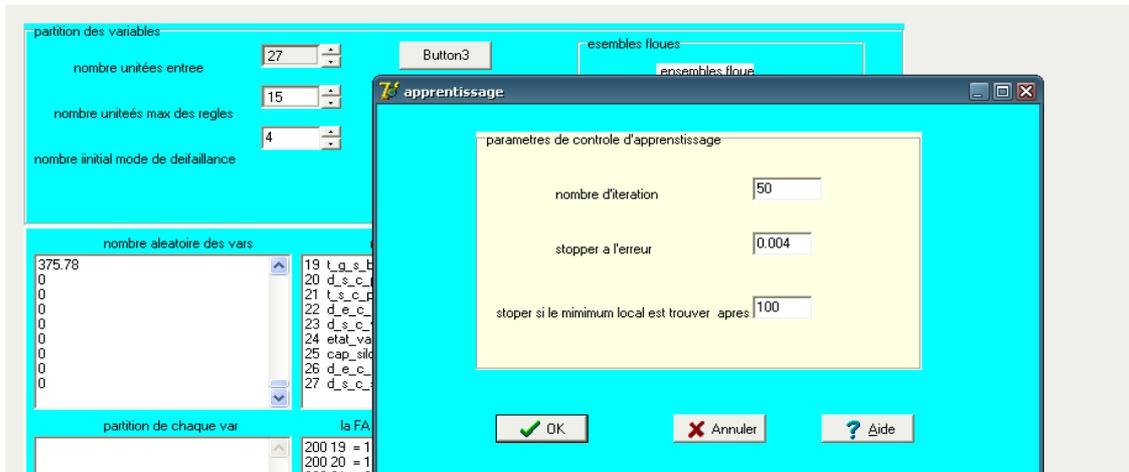


Figure 5.21 Les conditions d'apprentissage et de test [80]

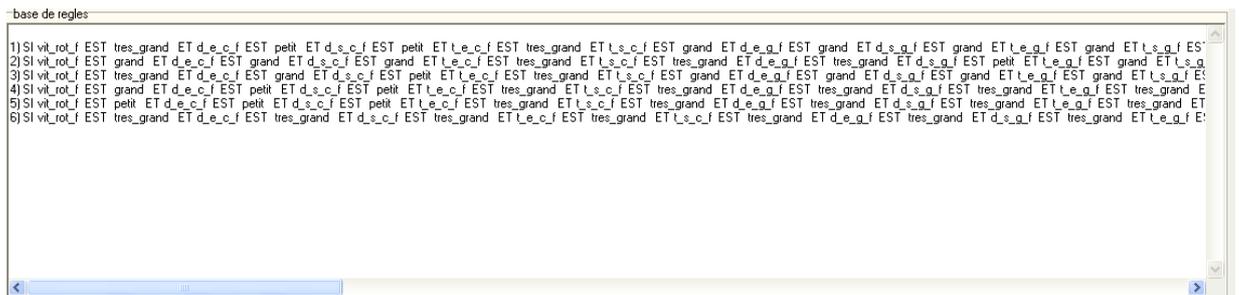


Figure 5.22 La base de règle Associé a une détection [72]

5.5.2 Résultats de simulation

Le nombre de clusters définit le nombre de règles floues. En appliquant la méthode de centre de moyenne floue [13,40] sur les données d'apprentissage et de vérifier la validité de la mesure proposée dans [13] il a été identifié qu'un nombre suffisant de clusters est de 4. Donc 4 règles floues ont été utilisées pour la base d'apprentissage et de raffinage. L'algorithme de catégorisation identifié les centres de cluster pour les données présentées [82] [84] [85].

- IF y (t-2) is A1 AND y (t-1) is B1 AND y(t) is C1 THEN y(t+1) is D1
- IF y(t-2) is A2 AND y(t-1) is B2 AND y(t) is C2 THEN y(t+1) is D2
- IF y(t-2) is A3 AND y(t-1) is B3 AND y(t) is C3 THEN y(t+1) is D3
- IF y(t-2) is A4 AND y(t-1) is B4 AND y(t) is C4 THEN y(t+1) is D4

La détection des Modes de défaillances dans l'atelier des SCIMAT clinker peut être affectée par un certain nombre de défauts. Ces défauts sont regroupés en quatre grandes catégories: les défauts de la tour de préchauffage, défauts de four rotatif, défauts refroidisseur et les défauts des ballons brûleur à gaz. Ici, seuls les défauts brusques ou initiaux sont pris en compte.

Faute	Description	Brusque/initial
F1	Chute de la jupe	I/A
F2	Bourrage	I/A
F3	Pas de break	I/A
F4	Transportateur à auget	I/A
F5	Presence anneaux	I
F6	Mauvaise homogénéisation	I/A
F7	Chute de croûtage	I/A
F8	Atteinte des briques réfractaires	I
F9	bourrage	I/A
F10	Moteur ventilateur tirage	I/A
F11	Courroies ventilateur tirage	I/A

Tableau 5.3 Description des fautes [88]

Les figures 5.23 et 5.24 montrent la réponse de la sortie du modèle normal et la production réelle de cinq à quinze minutes de prédiction et la figure 5.25 à 5.26 montrent la réponse de la sortie du modèle de faute et de la production réelle de trois à sept minutes de prédiction des données de test [85].

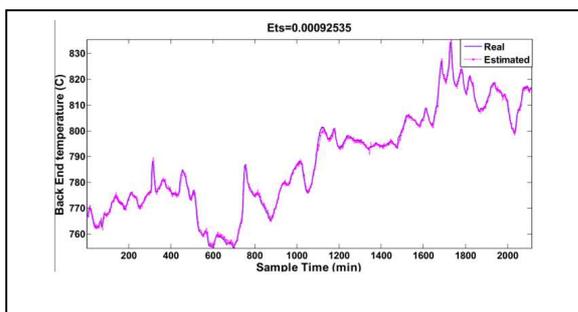


Figure 5.23 Modèle normal avec 5 min de prédiction [88]

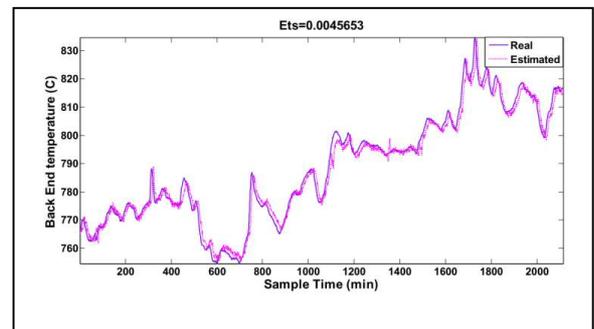


Figure 5.24 Modèle normal avec 10 min prédiction [88]

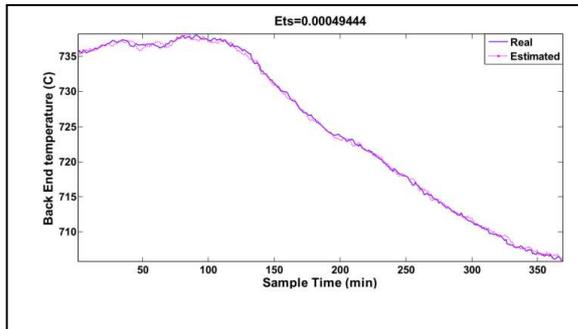


Figure 5.25 Le défaut du modèle avec 10 ou 15 min de prédiction [88]

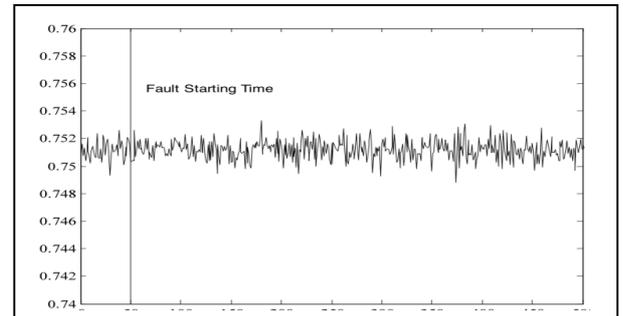


Figure 5.26 Effet de la panne F10 sur la rotation du four rotatif TPM [88]

5.6 Phase de diagnostic

Le diagnostic doit permettre à partir de l'observation de symptômes de remonter jusqu'aux causes expliquant ces symptômes. On considère donc qu'il existe une relation causale d'implication entre les causes et les effets observés : causes \rightarrow effets (symptômes). Cependant, la logique ne permet pas de fournir une information sur l'impliquant à partir de l'impliqué. L'idée de base développée ici est donc d'émettre une hypothèse explicative sur l'impliquant au regard des observations relatives à l'impliqué [18]. Appliquée au diagnostic, cela revient à émettre une hypothèse sur les causes en fonction de l'observation de symptômes.

Dans cette section nous nous sommes intéressés essentiellement à la réalisation de notre système d'aide au diagnostic, en passant par les étapes nécessaires telle que : le prétraitement des données, la présentation de ces derniers, la création (génération automatique) du système Neuro-Flou Temporel, son apprentissage et en fin les tests des résultats de cet apprentissage, leurs critiques et les solutions envisagées.

L'objectif de l'intégration des systèmes Neuro-Flous Temporels par rapport aux moyens existants est le fait de permettre une assistance efficace dans l'identification la plus précise possible des éléments défectueux. Il apparaît ainsi comme un support méthodologique et complémentaire aux aides dont disposent les techniciens de maintenance et les opérateurs de conduite. L'utilisateur fait appel au système après avoir constaté l'apparition des symptômes de pannes (messages, alarmes, voyant ...), le système de diagnostic oriente la recherche de la panne suivant une démarche exploitant les relations de causalité entre les dysfonctionnements de l'équipement.

Nous avons déjà présenté le fonctionnement de NEFSUR dans les sections précédentes, nous reprenons ici certains de ses aspects lors de son implantation dans le modèle simulé. Quand la recherche des causes de la défaillance élémentaire cause de dysfonctionnement globale est

trouvé (dans notre cas nous considérons une mauvaise clinkerisation comme défaillance majeure), NEFSUR informe l'utilisateur en déclenchant une alarme ayant deux composantes :

- un message décrivant la nature de la défaillance.
- Un cercle (ou flèche) indiquant la localisation du composant défaillant sur le schéma synoptique de l'installation.

Nous représentons sur les figures suivantes deux types de défaillances générées par l'opérateur et détectées par le système (figure 5.28 et figure 5.29).

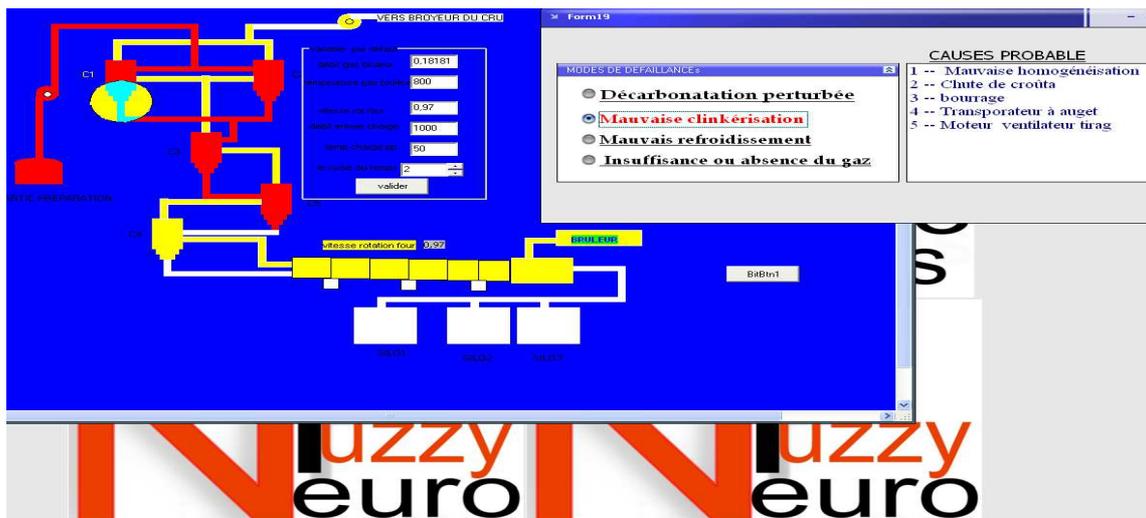


Figure 5.27 Exemple de message alarme (decarbonatation perturbé)

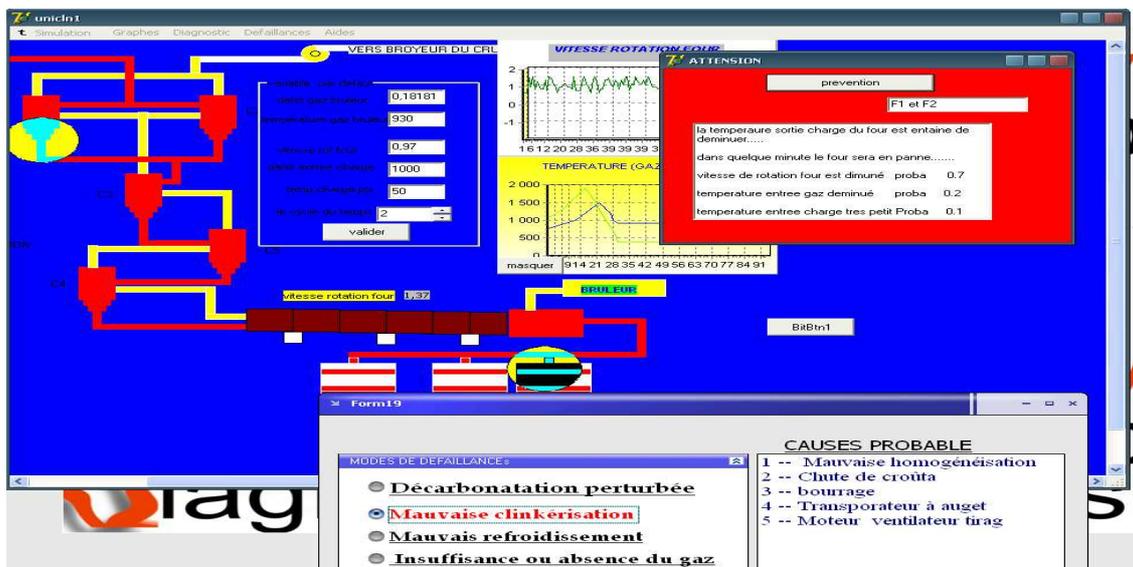


Figure 5.28 Exemple de message alarme (mauvaise clinkerisation)

5.6.1 Fonctionnement normal de l'unité de clinckersiation

La simulation du processus de clinkerisation en fonctionnement normal a permis de représenter l'atelier de clinkerisation en montant les étapes de passage de la charge et des gaz chauds dans les différents échangeurs .dans le système NFDIAG, les entrées du réseau sont les modes de défaillances avec leur criticité et degré d'appartenance des modes opératoires, et les sorties sont les causes, degré de crédibilité et la gravité .nous avons aussi extrait de l'AMDEC les données liées au four rotatif.

Modes de défaillance	cause	Fréquence	gravité
Décarbonatation perturbée (mauvaise)	• Chute de la jupe Bourrage	4	2
	• Pas de break	2	3
Mauvaise clinkerisation	• Transporateur à auget	2	4
	• Presence anneaux	5	3
	• Mauvaise homogénéisation	2	2
Mauvais refroidissement	• Chute de croûtage Atteinte des briques réfractaires	1	2
	• Bourrage	2	4
Insuffisance ou absence du gaz	• Moteur ventilateur tirage	2	2
	• Courroies ventilateur tirage	3	1

Tableau 5.3 AMDEC de la zone surveillée [Annexe 1]

L'idéal est de conduire le four à un niveau thermique inférieur et maintenir le reste de ses variables toujours autour de leurs consignes respectives (stabilité totale du reste de ses variables) à condition, bien sur que le niveau thermique minimum soit compatible avec une production de clinker de qualité. Le système de conduite peut recevoir des signaux mesurés des variables qui sont reportées sur le tableau 5.4

Il est évident qu'il existe une interrelation complexe entre toutes ces variables. Cependant, l'expérience montre qu'il est possible de commander le four en agissant sur certain nombre de variables jugées pertinentes. Ces variables pertinentes représentées sous formes d'attributs spécifiques des équipements afin de pouvoir simuler leurs variations et leurs effets sur la totalité de l'installation.

Variables avec signaux mesurés	Variables pertinentes commandées
Vitesse de rotation du four	Vitesse de rotation du four
Alimentation du cru	Alimentation du cru
Alimentation du combustible dans le Bruleur	Alimentation du gaz combustible
Concentration des oxydes d'azote(NOx) dans la sortie de circuit des gaz	Tirage du ventilateur principal
Concentration d'oxygène dans l'entrée au four	Aspiration des ventilateurs du refroidisseur
Pression dans le refroidisseur	
Pression dans la sortie du four	
Dépression dans l'entrée du four	
Concentration de monoxyde de carbone dans la sortie du circuit des gaz	
Débit des ventilateurs du refroidisseur	
Paire motrice du four	
Température des gaz dans le ventilateur principal	
Pyromètre dans la zone de clinkérisation	
Contenu en chaux libre du clinker	

Tableau 5.4 Variables liées au processus de clinkérisation

L'étape de simulation en fonctionnement normal les figures 5.29 et 5.30 illustrent trois variables critiques de l'unité étudiée qui sont : la vitesse de rotation du four, la température d'entrée gaz et la température de sortie charge du four dans un fonctionnement sans anomalies.

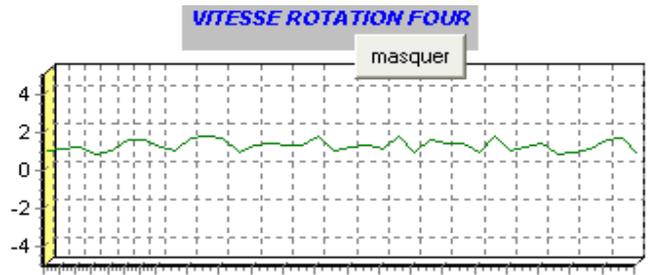


Figure 5.29 Vitesse de rotation four en mode normale

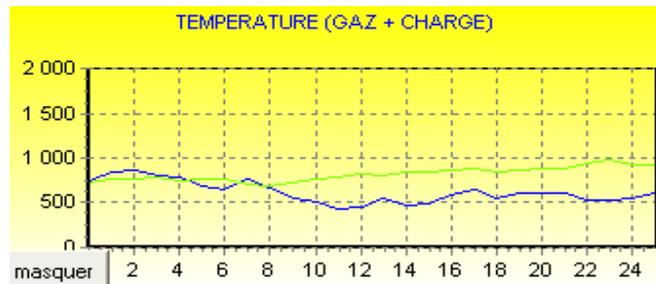


Figure 5.30 Température entrée charge et gaz du four en mode normale

5.6.2 Présence d'un dysfonctionnement

l'étape de simulation en fonctionnement avec la presence de dysfonctionnement les figures 5.31 et 5.32 illustres trois variables critique de l'unité étudié qui sont : la vitesse de rotation du four, le température entrée gaz et la température sortie charge du four dans un fonctionnement avec anomalies .

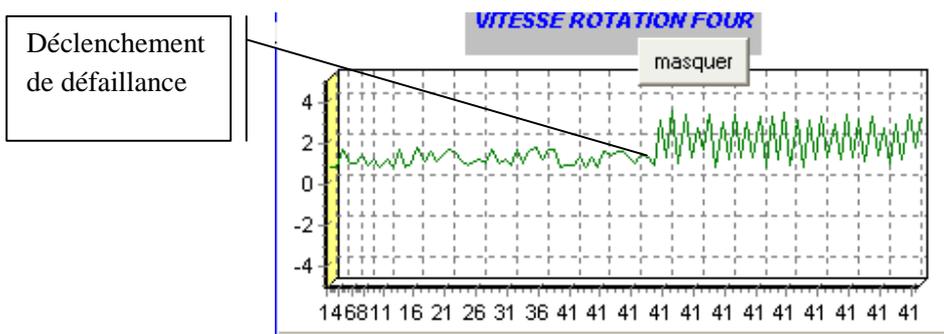


Figure 5.31 Vitesse de rotation four en présence d'une défaillance

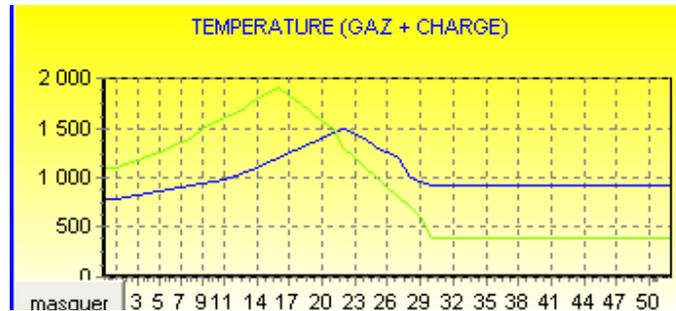


Figure 5.32 *Température de charge et de gaz, four en présence d'une défaillance*

5.7 Conclusion

Nous avons présenté dans ce chapitre l'intégration de l'outil d'aide à la surveillance. En partant de besoins simples (interfaçage avec outils et méthodes existants, intégration dans un ordinateur industriel ...), nous avons conçu un outil permettant de mettre en œuvre une surveillance dynamique d'un système industriel, capable de prévenir le responsable de maintenance en cas de défaillances et d'aider l'opérateur lors d'un diagnostic. Nous avons défini les outils dans les chapitres précédant.

Nous avons présenté un nouvel outil de diagnostic Neuro-Flou Temporel, nous avons détaillé la mise en œuvre d'un exemple d'application industrielle par l'outil de développement NEFSUR. Nous avons illustré d'utilisation de cet outil d'aide au diagnostic sous forme d'un prototype « NEFSUR » installé sur un PC.

Nous avons abordé les différentes étapes pour l'élaboration d'un système d'aide au diagnostic à partir de méthodes de classification et reconnaissances de formes floues. NEFSUR a été représenté comme un type spécial de Perceptron flou, à trois couches utilisé pour classifier des défaillances, en utilisant le système Neuro-Flou Temporel de type 3.

NEFSUR fait un apprentissage à deux phases :

Apprentissage des règles, NEFSUR génère les règles floues temporelles par le parcours des données ensuite optimise les règles par :

Apprentissage des paramètres des ensembles flous qui sont utilisés pour partitionner les données des formes à classer et les paramètres des données.

Conclusion générale

Conclusion générale et Perspectives

Les technologies pour la surveillance ont été largement développées ces dernières années. Elles se sont cependant traditionnellement concentrées sur la détection et l'isolement de défauts au niveau local. Actuellement, les chercheurs s'orientent vers de nouveaux concepts et l'activité de pronostic devient un élément clef des stratégies de maintenance industrielle. Par conséquent, la capacité à détecter et à isoler les défauts imminents, ainsi qu'à prévoir la situation future d'un système tendent à devenir des axes de recherches prioritaires. Il existe cependant encore peu de systèmes réels permettant de supporter ce type de processus. Les coûts élevés de maintenance des équipements complexes rendent nécessaire d'augmenter l'effort de développement d'outils support au "processus de surveillance".

D'une part, les travaux de recherche présentés dans cette thèse portent sur l'étude des systèmes Neuro-Flous Temporels pour la surveillance dynamique des systèmes de production industriels. Dans le cadre de la surveillance, notre étude concerne la détection de défaillances et le pronostic industriel. La surveillance classique a surtout tendance à traiter les variables d'une façon statique. Dans ce cas, la dynamique du système à surveiller n'est pas prise en compte, contrairement à la surveillance dynamique qui est capable par exemple de reconnaître une fausse alarme. Le pronostic quant à lui, correspond à la détection d'une dégradation avant que le système n'atteigne le seuil de défaillance.

Les principales contributions de cette étude sont regroupées en trois parties. Une première partie relative à l'état de l'art sur les différentes méthodologies de surveillance des systèmes de production. La deuxième partie de notre travail correspond à l'essentiel de notre contribution scientifique. En effet, nous avons proposé une nouvelle architecture des systèmes Neuro-Flous Temporels : l'architecture récurrente d'Elman. Nous avons développé un réseau de type Elman et l'élaborer pour accepter la compatibilité avec un système flou de type TSK. Pour la troisième partie nous avons développé un système d'aide au diagnostic par les systèmes Neuro-Flous, l'objectif de ce dernier est de chercher les causes probables d'une défaillance(s) par les méthodes de reconnaissance floue de formes.

Nous avons illustré l'utilisation de notre outil d'aide à la prédiction et de pronostic sous forme d'un prototype « NEFSUR » installer sur un PC.

NEFSUR est un système composé de deux modules : NEFPROG (Neuro-Fuzzy Prognosis) et NEFDIAG (Neuro-Fuzzy Diagnosis)

NEFPROG est un système Neuro-Flou Temporel qui permet de prédire les dégradations ou les défaillances futures d'un équipement en temps réel.

Quand le diagnostic est basé sur des observations multiples, ces dernières sont regroupées pour former des classes (modes de défaillance), auxquelles une nouvelle observation sera comparée pour être identifiée.

Travaux futurs

Ce travail a permis d'ouvrir les perspectives suivantes :

La limite que l'on peut rencontrer en utilisant un système Neuro-Flou Temporel (SNFT) est que généralement, pour une application donnée, la dimension du vecteur d'entrée d'un SNFT est a priori fixée. C'est donc là où peut résider toute la difficulté d'utiliser brutalement un SNFT pour la tâche de diagnostic car, en pratique, les informations utilisées pour le diagnostic sont souvent différentes (la dimension et la nature du vecteur d'entrée ne sont pas les mêmes pour différents diagnostics). On peut imaginer alors une solution distribuée, c'est-à-dire un réseau de neurones dynamique avec sa couche Neuro-Flou-Temporelle pour chaque type de cause. Chaque réseau de neurones possède bien évidemment son propre vecteur d'entrée qui caractérise le mieux l'identification de la cause de la défaillance (diagnostic). La décision globale du diagnostic final sera prise par un superviseur principal. Ce superviseur peut être établi soit à partir d'une

- base neuronale (un type de réseau de neurones – PMC ou RFR –),
- architecture des systèmes multi-agents.

Les systèmes multi-agents représentent des outils de l'intelligence artificielle avec certaines capacités très intéressantes concernant la prise de décision en fonction de certains critères imposés par l'expert. Cette architecture peut offrir une solution intéressante pour la prise de décision globale concernant le diagnostic en fonction des réponses locales de chaque architecture neuronale associée aux différentes causes.

Construire un système prédictif multi variables (la surveillance des systèmes complexes ne peut se baser sur le suivi d'un unique critère). De ce point de vue, l'architecture SNFT est a priori adaptée mais des essais supplémentaires doivent être envisagés,

Développer des systèmes Neuro-Flous de prédiction pouvant être interprétés sans que la cohérence des règles ne soit ambiguë.

Sur un plan plus général, plusieurs prolongements à ce travail sont envisageables et permettent de valider et d'améliorer l'approche proposée

Références Biobibliographiques

Références bibliographiques

- [1] AFNOR. Norme X60-319/NF EN 13306: *Terminologie de la maintenance*. AFNOR. Paris. 2001.
- [2] Y. El-shayeb. *Apport de la logique floue à l'évaluation de l'aléa . Mouvement de terrain des sites géométrique : proposition d'une méthodologie générale*. Thèse de doctorat, institut national polytechniques de Lorraine. 1999.
- [3] D. Racoceanu. *Contribution à la surveillance des systèmes de production en utilisant les techniques de l'intelligence artificielle*. Habilitation à diriger des recherches. Université de franche-Comté, France. 2006.
- [4] G. Zweingelstein *Diagnostic des défaillances; théories et pratique pour les systèmes industriels*, col. traite des nouvelles technologies, séries diagnostic et maintenance, Hermès. 1995.
- [5] M.R Zemouri. *Contribution à la surveillance des systèmes de production à l'aide des réseaux de neurones dynamiques : application à la maintenance*, Thèse de doctorat de l'université de Franche-Comté, France. 2003.
- [6] A. Villmeur, *Sûreté de fonctionnement des systèmes industriels fiabilité facteurs humains informatisation*. Edition eroylles. 1988.
- [7] C. Touzet, *Contribution à l'étude et au développement de modèle connexionniste séquentiel d'apprentissage*. Thèse de doctorat, Université de Montpellier, France. 1990.
- [8] M.S. Mouchaweh , *Conception d'un system de diagnostic adaptatif et prédictif basé sur la méthode fuzzy patern matching pour la surveillance en ligne des systèmes évolutifs Application à la supervision et au diagnostic d'une ligne de peinture au trempé*, thèse de doctorat, Reims chapagne-ardenne. 2002.
- [9] K. Khedhri, *Modélisation et simulation d'une application industrielle par le générateur de systèmes experts G2*, Thèse de Magister, Université de Batna Algerie, 2002.
- [10] S. Touaf, *Diagnostic logique des systèmes complexes dynamiques dans un contexte multi agent*. Thèse de Doctorat, Université Joseph Fourier, Grenoble, 2005.
- [11] L. Baghli, *Contribution à la commande de la machine asynchrone, utilisation de la logique floue, des réseaux de neurones et des algorithmes génétiques*. Thèse de doctorat, Université de Henri Poincaré, Nancy, France, 1999.

- [12] ISO : Norme ISO 13381-1 : *Surveillance et diagnostic des machines : Pronostic*. 2004.
- [13] I. Rasovska, *Contribution à une méthodologie de capitalisation des connaissances basée sur le raisonnement à partir de cas: Application au diagnostic dans une plateforme d'e-maintenance*. Thèse de doctorat, Laboratoire d'Automatique de Besançon (LAB-CNRS), Université de Franche-Comté, France, 2006.
- [14] O.E. Vasile, *Contribution au pronostic de défaillances par réseau Neuro-flou : maîtrise de l'erreur de prédiction*. Thèse de doctorat, UFR des sciences et techniques, Université de Franche-Comté, France, 2008.
- [15] N. Palluat , *Méthodologie de surveillance dynamique à l'aide des réseaux Neuro-flous temporels*. Thèse de doctorat, UFR des sciences et techniques, Université de Franche-Comté, France, 2006.
- [16] M.D Mouss. *Diagnostic et conduite des systèmes de production par approche base de connaissances*. Thèse de doctorat, Laboratoire d'Automatique de productique, Université de Batna, Algérie 2006.
- [17] Van Tung Tran, *Machine Fault Diagnosis and Condition Prognosis using Adaptive Neuro-Fuzzy Inference System and Classification and Regression Trees*. Thèse de doctorat , Department of Mechanical Engineering The Graduate School Pukyong National University. 2009.
- [18] D.Hedjazi, *Conception d'un modèle coopératif de support de la télémaintenance industrielle*. Thèse de doctorat, Université de Batna, Algérie, 2011.
- [19] L. Hartert, *Reconnaissance des formes dans un environnement dynamique appliquée au diagnostic et au suivi des systèmes évolutifs*. Thèse de doctorat, l'UFR sciences exactes et naturelles, université de Reims Champagne-Ardenne, France, 2010.
- [20] A. Mokhtari, *Diagnostic des systèmes hybrides : développement d'une méthode associant la détection par classification et la simulation dynamique*. Thèse de doctorat, l'Université de Toulouse, France, 2007.
- [21] T. Kempowsky, *Surveillance de procédés a base méthodes de classification : conception d'un outil d'aide pour la détection et le diagnostic des défaillances*. Thèse de doctorat, l'Université de Toulouse, France, 2004.
- [22] P. Cocheteux, *Contribution a la maintenance proactive par la formalisation du processus de pronostic des performances des systèmes industriels*. Thèse de doctorat, l'université de henry Poincaré, Nancy, France, 2010.
- [23] E. R. Loures, *Surveillance et diagnostic des phases transitoires des systèmes hybrides bases sur l'abstraction des dynamiques continues par réseau de pétri temporel flou*. Thèse de doctorat, université de Sabatier de Toulouse, France, 2006.

- [24] P. Ribot, *Vers l'intégration diagnostic, pronostic pour la maintenance des systèmes complexes*. Thèse de doctorat, université de Toulouse, France, 2009.
- [25] J. Biteus, *Distributed diagnosis and simulation based residual generators*. Thèse de doctorat, université de Linköpings, Suède, 2005.
- [26] N. Zenati-Henda, *Contribution à la conception et à la réalisation d'un système de réalité augmentée pour la maintenance*. Thèse de doctorat, L'UFR des Sciences et Techniques De l'Université de Franche-Comté, France, 2008.
- [27] V. Cocquempot, *Contribution à la surveillance des systèmes industriels complexes*. Habilitation à diriger des recherches. Laboratoire d'Automatique et de Génie Informatique et Signal de Lille, France, 2006.
- [28] O. Ondel, *Diagnostic par reconnaissance des formes : application à un ensemble convertisseur – machine asynchrone*. Thèse de doctorat, l'école centrale de Lyon, France, 2006.
- [29] R. Casimir, *Diagnostic des défauts des machines asynchrones par reconnaissances de formes*. Thèse de doctorat, l'école centrale de Lyon, France, 2004.
- [30] M. Knotek, *Fault diagnostics based on temporal analysis*. Thèse de doctorat, l'université Joseph Fourier, Grenoble, France, 2006.
- [31] R. Alaoui, *Conception d'un module de diagnostic à base des suites de bandes temporelles en vue de la supervision des procédés énergétique. Application en ligne à un Générateur de vapeur*. Thèse de doctorat, Université des Sciences et Technologies de Lille, France, 2006.
- [32] K. Medjaher. *Contribution de l'outil bond graph pour la Conception de systèmes de supervision des processus Industriels*. Thèse de doctorat, Université des Sciences et Technologies de Lille, France, 2005.
- [33] M. Orchard. *A Particle Filtering-based Framework for On-line Fault Diagnosis and Failure Prognosis*. HDR, School of Electrical and Computer Engineering Georgia Institute of Technology, USA, 2005.
- [34] M. Witczak. *Fault detection of nono-linear dynamical systems*. Lecture Notes in Control and Computer Science Volume 1. Faculty of Electrical Engineering, Computer Science and Telecommunications University of Zielona Góra, 2003.
- [35] A. Zerguy. *Décision collaborative dans les systèmes distribués – application à la maintenance*. Thèse de doctorat, l'Institut National Polytechnique de Toulouse, France, 2008.
- [36] S. Chekroun. *Commande neuro-floue sans capteur de vitesse d'une machine asynchrone triphasée*. Mémoire de magister, Ecole Normale Supérieure d'Enseignement Technologique d'Oran, Algérie, 2009.

- [37] Wilson Q. Wang. *Prognosis of machine health condition using neuro-fuzzy systems*. Mechanical Systems and Signal Processing 18 pp. 813–831, Elsevier, 2003.
- [38] M. El koujok. *Contribution au pronostic industriel : intégration de la confiance à un modèle prédictif neuro-flou*. Thèse de doctorat, l’U.F.R des sciences et techniques de l’université de Franche-Comté, France, 2010.
- [39] G.Vachtsevanos, F. Lewis, M. Roemer, A. Hess, B. Wu. *Intelligent fault Diagnosis and prognosis for engineering systems*. John wiley & sons, inc. 2006.
- [40] S Simani, F. Cesare and Ron J. Patton. *Model-based fault diagnosis in dynamic systems using identification techniques*. Springer-Verlag, 2002.
- [41] E Christopher, T Lombae, and Hafid Smaili. *Fault Tolerant Flight Control: A Benchmark Challenge*. Lecture Notes in Control and Information Sciences .2010.
- [42] N. Arzu. *A temporal neuro-fuzzy approach for time series analysis*. Thèse de doctorat, the graduate school of natural and applied sciences Of the middle east technical university, 2003.
- [43] L. Shen , E.H. Tay , b Liangsheng Qu , Y. Shen . *Fault diagnosis using Rough Sets Theory*, Computers in Industry 43, pp 61–72, Elsevier, 2000.
- [44] R. Isermann . *Model-based fault detection and diagnosis - status and applications -*. Institute of Automatic Control, Darmstadt University of Technology, IFAC, 2004.
- [45] V. Venkatasubramanian, R. Rengaswamy, *A review of process fault detection and diagnosis Part III: Process history based methods*. Computers and Chemical Engineering 27 pp 327 / 346, Elsevier, 2003.
- [46] M. J. Watts. *ANN Rule Extraction using Evolutionary Programmed Fuzzy Membership Functions*. International Journal of Information Technology Vol. 11 No. 10, pp45-53, Singapore Computer Society , 2005.
- [47] R. Zafarani , M. R. Yazdchi. *A Novel Action Selection Architecture in Soccer Simulation Environment Using Neuro-Fuzzy and Bidirectional Neural Networks*. International Journal of Advanced Robotic Systems, Vol. 4, No.1, 2007.
- [48] E. Marcos, Orchard, j. George. *A particle-filtering approach for on-line fault diagnosis and failure prognosis*. Transactions of the Institute of Measurement and Control 31, 3/4 pp. 221–246, 2009.
- [49] G. Betta, C Liguori, A. Paolillo, A. Pietrosanto. *A DSP-Based FFT-Analyzer for the Fault Diagnosis of Rotating Machine Based on Vibration Analysis*. IEEE transactions on instrumentation and measurement, vol. 51, no. 6. pp 162-175, december, 2002 .
- [50] R. Isermann. *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Springer-Verlag Berlin Heidelberg, 2006.

- [51] O. Dragomir, R. Gouriveau, N. Zerhouni. *Pronostic industriel : étude de l'erreur de prédiction du système ANFIS*. Automatique Avancée et Informatique Appliquée. 1, 1, PP 99-105 . 2009.
- [52] F. Aminian, M. Aminian. *Analog Fault Diagnosis of Actual Circuits Using Neural Networks*. IEEE transactions on instrumentation and measurement, vol. 51, no. 3, pp 320-332, 2002.
- [53] Y. Tharrault, G. Mourot, J. Ragot, D. Maquin . *Fault detection and isolation with robust principal component analysis* .Int. J. Appl. Math. Comput. Sci., , Vol. 18, No. 4, pp 429–442, 2008.
- [54] P. Krzysztof. *Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes*. Lecture Notes in Control and Information Sciences377. Springer-Verlag Berlin Heidelberg, 2008.
- [55] P. Cocheteux, A. Voisin, E. Levrat et B. Iung. *Formalisation du pronostic à base d'une approche processus*. 3ème Colloque International Francophone Performance et Nouvelles Technologies en Maintenance, PENTOM 2007, Mons, Belgium, 2007.
- [56] A. Soukkou ,A. Khellaf , S. Leulmi. *Supervision neuro-floue à apprentissage génétique d'un PID robuste*. SCIENCES & Technologie B– n°23, pp. 95-106, 2005.
- [57] B. Zhang, G. Georgoulas. *Rolling Element Bearing Feature Extraction and Anomaly Detection Based on Vibration Monitoring*.16th Mediterranean Conference on Control and Automation .Congress Centre, Ajaccio, France .25-27 June, 2008.
- [58] W. Douglas, L. Derek, George Georgoulas. *Real-Time Fault Detection and Accommodation for COTS Resolver Position Sensors*. International conference on prognostics and health management, begin, China, 6-9 Oct. 2008
- [59] B. Zhang, C. Sconyers, C. Byington, R. Patrick. *Anomaly Detection: A Robust Approach to Detection of Unanticipated Faults*. International conference on prognostics and health management. 2008.
- [60] V. Venkatasubramanian,R. Rengaswamy, S. Kavuri,K. Yin. *A review of process fault detection and diagnosis Part III: Process history based methods*. Computers and Chemical Engineering 27 PP. 327 / 346, Elsevier ,.2003.
- [61] R. Olsson .*Batch Control and Diagnosis*.Department of Automatic Control. Lund Institute of Technology, Lund, Sweden. June 2005.
- [62] M. Orchard, g. Vachtsevanos. *A Particle Filtering Approach for On-Line Fault Diagnosis and Failure Prognosis* . the Transactions of Institute of Measurement and Control, 2007.

- [63] D. Nauck , F. kalaaounn. *Neuro-Fuzzy classification initialized by fuzzy clustering*. The forth European congress of intelligent technique and soft computing, aicen, Sweden, 1996.
- [64] D. Nauck. *Data Analysis with Neuro-Fuzzy Methods*. HDR, Université de Magdeburg, 2000.
- [65] U. Nauck . *Design and Implementation of a Neuro-Fuzzy Data Analysis Tool in Java*. In BT Technology Journal, Vol. 16, No. 3. July 1998.
- [66] S. Mitra, Y. Hayashi. *Neuro–Fuzzy Rule Generation: Survey in Soft Computing Framework*. IEEE transactions on neural networks, vol. 11, no. 3.pp 748-768, May 2000.
- [67] S. Dennis. S. Lee, J. Lithgow, R. Morrison. *New Fault Diagnosis of Circuit Breakers*. IEEE transactions on power delivery, vol. 18, no. 2. Pp 454-459, April 2003.
- [68] E. Pasha. *Fuzzy Linear Regression Models with Fuzzy Entropy*. Applied Mathematical Sciences, Vol. 1, no. 35, 1715 – 1724.1716-1724, 2007.
- [69] W. Martin. Liddament. *Real Time Simulation of HVAC Systems for Building Optimization, Fault Detection and Diagnostics*. IEA Energy Conservation in Buildings & Community Systems Programme, 2001.
- [70] M. Basseville, M. Cordier. *Surveillance et diagnostic de systèmes dynamiques: approches complémentaires du traitement de signal et de l'intelligence artificielle*. rapport de recherche. Institut national de recherche en informatique et en automatique, 1999.
- [71] G. Levitin. *Computational Intelligence in Reliability Engineering: New Metaheuristics, Neural and Fuzzy Techniques in Reliability*. Springer-Verlag Berlin Heidelberg, 2007.
- [72] R. Mahdaoui, H. Mouss and D. Mouss .*Industrial dynamics monitoring by Temporals Neuro-Fuzzy systems: Application to manufacturing system*. ICGST International Journal on Graphics, Vision and Image Processing (GVIP) <http://www.icgst.com/gvip/journal/Issues.html>..2009.
- [73] R. Mahdaoui, H. Mouss and Djamel Mouss. *NEFDIAG, a New Approach for Industrial diagnosis by Neuro-Fuzzy systems: Application to manufacturing System*. european ideas for busnis. Vol 2, num 2 .2008.
- [74] R. Mahdaoui, H. Mouss et Ouhiba Chouhal . *Reconnaissances dynamique de formes par les systèmes Neuro Flous temporels : Application au diagnostic Industriel*. SNIB,

- 6eme Séminaire Nationale en Informatique de Biskra, Algérie du 06 au 08 Mai 2008.
- [75] R. Mahdaoui, H. Mouss . *Diagnostic industriel par approche Neuro-floue application a un système de production*, la conférence internationale en productique, setif, 3-4 novembre, 2007.
- [76] R. Mahdaoui, H. Mouss, Dajmel Mouss et Chouhal Ouahiba .*Hybridation des systèmes Neuro Flous et la recherche Tabou pour la reconnaissance des formes*. JSIA'09, Journées Scientifiques sur l'Informatique et ses applications. Guelma, Algérie du 03 au 04 Mars 2009.
- [77] R. Mahdaoui, Hayet Mouss et Djamel Mouss . *Surveillance Industrielle dynamique par les systèmes Neuro Flous Temporels : Application à un système de Production*. J2I'2009, Journée d'Instrumentation Industrielle, Université d'Oran, le 18 Avril 2009.
- [78] R. Mahdaoui, H. Mouss O. Chouhal . *Reconnaissances de formes par les systèmes Neuro-Flous temporels : Application au diagnostic Industriel*. MCSEAI, Magheblian Conference on Information Technologies. Oran, Algérie 28 au 30 Avril 2008.
- [79] R. Mahdaoui, H. Mouss et D. Mouss, «*NEFDIAG, un outil d'aide au diagnostic industriel par Neuro-Flou: Application à un système de production*. CIFA'2008 Conférence Internationale Francophone d Automatique, Bucarest, Roumanie, 03 au 05 Septembre 2008.
- [80] R. Mahdaoui, H. Mouss et O. Chouhal. *Surveillance industrielle dynamique par les systèmes Neuro-Flou: Application à un système de production*. CIFA'2008 Conférence Internationale Francophone d Automatique, Bucarest, Roumanie, 03 au 05 Septembre 2008.
- [81] R. Mahdaoui, H. Mouss et D. Mouss. *Surveillance industrielle dynamique par les systèmes Neuro-Flou: Application à un système de production*. QUALITA 2009, 8ème édition du congrès international pluridisciplinaire – Besançon, France du 18 au 20 mars 2009.
- [82] O. Chouhal, R. Mahdaoui, H. Mouss, H. Houassi . *Problème d'apprentissage des systèmes Neuro Flous numériques et ou Symboliques pour la reconnaissance dynamique de formes : Application au diagnostic Industriel*. QUALITA, 8ème édition du congrès international pluridisciplinaire – Besançon France du 18 au 20 mars 2009.
- [83] R. Mahdaoui, H. Mouss et D. Mouss .*Surveillance industrielle dynamique par les systèmes Neuro-Flou: Application à un système de production* .SETIT'09, Sciences of Electronics, Technologies of Information and Telecommunication, Hammamet Tunisie, du 22 au 26 Mars 2009.
- [84] R. Mahdaoui , H. Mouss , D. Mouss, O. Chouhal . *Hybridation des systèmes Neuro Flouset la recherche Tabou pour la reconnaissance des formes*. ICISP'09

- International Conference on Systems and Processing Information Guelma, Algérie du 02 au 04 Mai 2009.
- [85] R. Mahdaoui , H. Mouss , D. Mouss, O. Chouha . *Surveillance industrielle dynamique par les systèmes Neuro-Flous Temporels : Application a un système de production*. ICISP'09 International Conference on Systems and Processing Information Guelma, Algérie du 02 au 04 Mai 2009.
- [86] H. Mouss, R. Mahdaoui, D. Mouss, N. Mouss . *Industrial Diagnosis by Fuzzy systems : Application to Manufacturing System* . IECONE17, 17th International Conference en Nuclear Engineering, Brussels Belgium, du 15 au 17 July 2009.
- [87] R. Mahdaoui , H. Mouss . *Les systèmes Neuro Flous comme outil de reconnaissances statiques/dynamiques des formes*. ISPA'2009 First Internationale Conference on Image and Signal processing, Mostaganem.2009.
- [88] R. Mahdaoui , H. Mouss , D. Mouss, O. Chouha . *Reconnaissance des formes par les systèmes neuro-flous ELMAN: Application au pronostic des systèmes de production*. ICIEM'10 International Conference on Industrial Engineering and Manufacturing. Batna–Algeria du 09 au 10 Mai 2010.
- [89] R. Mahdaoui , H. Mouss , D. Mouss, O. Chouhal . *Apprentissage des systèmes Neuro-Flous par des données numériques et/ou symboliques pour la reconnaissance dynamique de formes : Application au diagnostic industriel*. ICIEM'10 International Conference on Industrial Engineering and Manufacturing. Batna–Algeria du 09 au 10 Mai 2010.
- [90] O. Kadri, H. Mouss, A. Abdelhdi, et R. Mahdaoui . *Vers une optimisation de l'algorithme anttreestoch*. ICIEM'10 International Conference on Industrial Engineering and Manufacturing. Batna, Algeria du 09 au 10 Mai 2010.
- [91] R. Mahdaoui, H. Mouss et D. Mouss. *Fault prognosis by temporal Neuro-Fuzzy systems: application for manufacturing systems*. SETIT'12, Sciences of Electronics, Technologies of Information and Telecommunication, Hammamet Tunisie, du 21 au 24 Mars 2012.
- [92] R. Mahdaoui, H. Mouss et D. Mouss. *A Temporal Neuro-Fuzzy Monitoring System to Manufacturing Systems*. IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 1, pp 237-246, May 2011 ISSN (Online): 1694-0814.2011.
- [93] R. Mahdaoui, H. Mouss et D. Mouss. *Temporal Neuro-Fuzzy systems in fault diagnosis and prognosis*. International review on modeling ans simulations (IREMOS)., vol.4 N.1, pp 436-440, February 2011, ISSN1974-9821, 2011.

- [94] R. Mahdaoui. *Diagnostic industriel par approche Neuro-floue*, mémoire de magister, laboratoire d'automatique et productique, université de Batna, 2008.
- [95] S. Benaïcha. *Le pronostic industriel par les systèmes Neuro-Flous : Application à un système de production*. mémoire de magister, laboratoire d'automatique et productique, université de Batna, 2012.

Liste des publications personnelles

Liste des publications personnelles

Publications dans des revues internationales

1. **R. Mahdaoui**, H. Mouss et D. Mouss. A Temporal Neuro-Fuzzy Monitoring System to Manufacturing Systems. IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 1, pp 237-246, May 2011 ISSN (Online): 1694-0814.2011.
2. **R. Mahdaoui**, H. Mouss et D. Mouss. Temporal Neuro-Fuzzy systems in fault diagnosis and prognosis. International review on modeling and simulations (IREMOS)., vol.4 N.1, pp 436-440, February 2011, ISSN1974-9821, 2011.
3. **R. Mahdaoui**, H. Mouss A TSK-Type Recurrent Neuro-Fuzzy Systems for Fault Prognosis. Journal of Software Engineering and Applications. Vol. 58, Issue 7, PP.449-458. ISSN Print: 1945-3116 ,ISSN Online: 1945-3124 . 2012
4. R. Mahdaoui, H. Mouss et *M. D. Mouss, O. Chouhal*. The Temporal Neuro-Fuzzy Systems Learning Using Artificial Immune Algorithm. International Review of Mechanical Engineering (IREME). vol.4 N.1, pp 918-922, May 2012, ISSN 1970 - 8734, 2012

Communications dans des conférences internationales

5. **R. Mahdaoui**, H. Mouss and D. Mouss, Industrial dynamics monitoring by Temporals Neuro-Fuzzy systems: Application to manufacturing system. ICGST International Journal on Graphics, Vision and Image Processing (GVIP) <http://www.icgst.com/gvip/journal/Issues.html>..2009.
6. **R. Mahdaoui**, H. Mouss and Djamel Mouss, NEFDIAG, a New Approach for Industrial diagnosis by Neuro-Fuzzy systems: Application to manufacturing System. europen ideas for busnis. Vol 2, num 2,2008.
7. **R. Mahdaoui**, H. Mouss, Diagnostic industriel par approche neuro-floue application a un système de production, Conférence Internationale en Productique, Sétif, 3-4 novembre, 2007.

8. **R. Mahdaoui**, H. Mouss O. Chouhal, Reconnaissances de formes par les systèmes neuro-flous temporels : Application au diagnostic Industriel, MCSEAI, Maghebian Conference on Information Technologies. Oran, Algérie 28 au 30 Avril 2008
9. **R. Mahdaoui**, H. Mouss et D. Mouss, «NEFDIAG, un outil d'aide au diagnostic industriel par neuro-flou: Application à un système de production, CIFA'2008 Conférence Internationale Francophone d'Automatique, Bucarest, Roumanie, 03 au 05 Septembre 2008.
10. **R. Mahdaoui**, H. Mouss et O. Chouhal, Surveillance industrielle dynamique par les systèmes neuro-flou: Application à un système de production, CIFA'2008 Conférence Internationale Francophone d'Automatique, Bucarest, Roumanie, 03 au 05 Septembre 2008.
11. **R. Mahdaoui**, H. Mouss et D. Mouss, Surveillance industrielle dynamique par les systèmes neuro-flou: Application à un système de production, QUALITA 2009, 8ème édition du congrès international pluridisciplinaire – Besançon, France du 18 au 20 mars 2009.
12. **R. Mahdaoui**, H. Mouss et D. Mouss, Surveillance industrielle dynamique par les systèmes neuro-flou: Application à un système de production, SETIT'09, Sciences of Electronics, Technologies of Information and Telecommunication, Hammamet Tunisie, du 22 au 26 Mars 2009.
13. **R. Mahdaoui**, H. Mouss, D. Mouss, O. Chouhal, Hybridation des systèmes Neuro Flouset la recherche Tabou pour la reconnaissance des formes. ICISP'09 International Conference on Systems and Processing Information Guelma, Algérie du 02 au 04 Mai 2009.
14. **R. Mahdaoui**, H. Mouss, D. Mouss, O. Chouha, Surveillance industrielle dynamique par les systèmes neuro-flous Temporels : Application a un système de production, ICISP'09 International Conference on Systems and Processing Information Guelma, Algérie du 02 au 04 Mai 2009.
15. H. Mouss, **R. Mahdaoui**, D. Mouss, N. Mouss, Industrial Diagnosis by Fuzzy systems : Application to Manufacturing System, IECONE17, 17th International Conference en Nuclear Engineering, Brussels Belgium, du 15 au 17 July 2009.
16. **R. Mahdaoui**, H. Mouss, Les systèmes Neuro Flous comme outil de reconnaissances statiques/dynamiques des formes. ISPA'2009 First International Conférence on Image and Signal processing, Mostaganem.2009.
17. **R. Mahdaoui**, H. Mouss, D. Mouss, O. Chouha, Reconnaissance des formes par les systèmes neuro-flous ELMAN: Application au pronostic des systèmes de production, ICIEM'10 International Conférence on Industrial Engineering and Manufacturing. Batna–Algeria du 09 au 10 Mai 2010.

18. **R. Mahdaoui** , H. Mouss , D. Mouss, O. Chouhal, Apprentissage des systèmes neuro-flous par des données numériques et/ou symboliques pour la reconnaissance dynamique de formes : Application au diagnostic industriel, ICIEM'10 International Conference on Industrial Engineering and Manufacturing. Batna–Algeria du 09 au 10 Mai 2010.
19. O. Kadri, H. Mouss, A. Abdelhdi, et **R. Mahdaoui**, Vers une optimisation de l'algorithme anttreestoch, ICIEM'10 International Conference on Industrial Engineering and Manufacturing. Batna, Algeria du 09 au 10 Mai 2010.
20. **R. Mahdaoui**, H. Mouss et D. Mouss, Fault prognosis by temporal Neuro-Fuzzy systems:application for manufacturing systems. SETIT'12, Sciences of Electronics, Technologies of Information and Telecommunication, Hammamet Tunisie, du 21 au 24 Mars 2012.

Communications dans des congrès nationaux

21. **R. Mahdaoui**, Hayet Mouss et Djamel Mouss, Surveillance Industrielle dynamique par les systèmes Neuro Flous Temporels : Application à un système de Production. J2I'2009, Journée d'Instrumentation Industrielle, Université d'Oran Es Sénia, le 18 Avril 2009.
22. **R. Mahdaoui**, H. Mouss, Dajmel Mouss et Chouhal Ouahiba,Hybridation des systèmes Neuro Flous et la recherche Tabou pour la reconnaissance des formes, JSIA'09, Journées Scientifiques sur l'Informatique et ses applications. Guelma, Algérie du 03 au 04 Mars 2009.
23. **R. Mahdaoui**, H. Mouss et Ouhiba Chouhal, Reconnaissances dynamique de formes par les systèmes Neuro Flous temporels : Application au diagnostic Industriel, SNIB 2008, 6eme Séminaire Nationale en Informatique de Biskra, Algérie du 06 au 08 Mai 2008.

Mémoire

24. **R. Mahdaoui**, Diagnostic industriel par approche neuro-floue, mémoire de magister, laboratoire d'automatique et productique, université de Batna, 2008.

Annexes

LAP	ANALYSE DES MODES DE DEFAILLANCE DE LEURS EFFETS ET DE CRITICITE									PROCESSUS AMDEC	
	composant	fonction	Mode de défaillance	repère	Causes possibles	Durée indisponibilité	Effet sur le système	Criticité	Information de l'opérateur		Activité de compensation de l'opérateur face à la situation
Tour de préchauffage	Préparer la matière pour la cuisson de la matière (Décarbonatation à contre courant)	Décarbonatation perturbée (mauvaise)	1	Chute de la jupe	2 h	Affecte la qualité, la production et la consommation d'énergie	18	Alarme pression température Analyseur (augmentation CO, diminution O2)	l'opérateur agit sur les variables de commande: débit entrée farine, débit gaz, vitesse four, vitesse ventilateur tirage pour compenser sinon l'opérateur appel le surveillant pour identifier le problème et intervenir,		
			2	Bourrage	40 mn					Arrêt du four	
Four rotatif	Acheminement matière/gaz chauds Et clinkérisation	Mauvaise clinkérisation	3	No break	2 h	Affecte la qualité, la production et la consommation d'énergie	20	Arrêt complète (usine)	Aucune intervention (démarrage du groupement électrogène pour virer le four) Arrêt du four		
			4	Transporteur à auget	12 h					16	Alarme
			5	Présence anneaux	1 h					30	Perturbation de la chaux libre, température clinker, moment du four, poids au litre, consommation calorifique spécifique
			6	Mauvaise homogénéisation	24 h					32	Perturbation de la chaux libre,

			7	Chute de croûtage	24 h		24	température clinker, moment du four, poids au litre	
			8	Atteinte des briques réfractaires	12 h		16	Perturbation du moment du four, augmentation du poids au litre, augmentation de la chaux libre Augmentation de la température virole	Injection d'eau Diminuer vitesse four Diminuer vitesse four Arrêt du four
Refroidisseur à ballonnets	Refroidissement rapide du clinker	mauvais refroidissement	9 10	Bourrage Courroies ventilateur tirage	2h 3h	Affecte la qualité, la production et la consommation d'énergie	12 12	Augmentation pression air secondaire Alarme	Arrêt du four Arrêt du four
Brûleur à gaz	Fournir le gaz de combustion	Insuffisance ou absence	11 12	Courroies ventilateur tirage Moteur ventilateur tirage	3h 2h	Affecte la qualité, la production et la consommation d'énergie	12 12	Perturbation des température et Pression tour et électrofiltre	Arrêt du four Arrêt du four

Dans cette annexe nous présentons quatre algorithmes d'apprentissages en utilisant les notions suivantes:

- Γ : un ensemble de données d'apprentissage avec $|\Gamma| = s$, qui représente un problème de classification de défaillances, tel que la forme $p \in \mathfrak{R}^n$ qui doivent être classées dans m mode de défaillances, tel que $C_i \subseteq \mathfrak{R}^n$.
- $(p, t) \in \Gamma$: une forme d'apprentissage consiste d'un vecteur d'entrée $p \in \mathfrak{R}^n$ et un vecteur de sortie (désiré) $t \in [0, 1]^m$. L'index de la classe de \mathbf{P} est calculé par le grand valeur de \mathbf{t} : classe (p) = $\text{argmax}_j \{t_j\}$.
- $R = (A, C)$: une règle floue de classification avec un antécédent $\text{ant}(R) = A$ et conséquent $\text{con}(R) = C$, avec $A = (\mu_{j_1}^{(1)}, \dots, \mu_{j_n}^{(n)})$ et C est un classe (mode de défaillance). Le degré d'accomplissement d'une règle R pour une forme p est $R(p) = A(p) = \min\{\mu_{j_1}^{(1)}(p_1), \dots, \mu_{j_n}^{(n)}(p_n)\}$.
- $\mu_j^{(i)}$ est la j^{eme} ensemble floue de partition floue initial du variable d'entée x_i .

Il existe q_i ensemble flou pour le variable x_i .

- \mathbf{c}_A : un vecteur de m entrées pour présenter les degrés d'appartenance cumulées pour chaque classe de toutes les formes avec $A(p) > 0$; $\mathbf{c}_A[j]$ est le j^{eme} entrée de \mathbf{c}_A .
- $P_R \in [-1, 1]$ une valeur qui représente la performance de règle R :

$$P_R = \frac{1}{s} \sum_{(p,t)} (-1)^c R(p), \text{ avec } c = \begin{cases} 0 & \text{si } \text{calssse}(p) = \text{con}(R), \\ 1 & \text{si autrement} \end{cases}$$

- $\nu_r^{(j)}$: l'ensemble floue du variable de sortie y_j ($j \in \{1, \dots, m\}$) qui est incluse dans le conséquent de règle floue R_r .
- C_j : le vecteur de sortie qui enregistre le degré d'appartenance de forme d'entrée pour la classe j .

Algorithme 1. Algorithme d'apprentissage de règles par NEFDIAG.

Début algorithme

Pour toutes forme (p,t) **faire**

Pour toutes entrée x_i **faire**

$$\mu_{ji}^{(i)} = \arg \max_{\mu_j^{(i)}, j \in \{1, \dots, q_i\}} \{ \mu_j^{(i)}(p_i) \}$$

Fin pour

 Créer l'antécédent $A = (\mu_{j1}^{(1)}, \dots, \mu_{jn}^{(n)})$;

Si ($A \notin$ liste des antecedents) **alors**

 Ajouter A da, s la liste des antecedents.

Finsi

Finpour

Pour toutes forme (p,t) **faire**

Pour toutes $A \in$ liste d'antécédents **faire**

$$c_A[\text{classe}(p)] = c_A[\text{classe}(p)] + A(p)$$

finpour

finpour

pour toutes $A \in$ liste d'antécédents **faire**

$$j = \arg \max_{i \in \{1, \dots, m\}} \{ c_A[i] \} ;$$

 Créer une règle R avec l'antécédent A et conséquent C_j ;

 Ajouter R dans la base de règle candidates ;

$$P_R = \frac{1}{s} c_A[j] - \sum_{i \in \{1, \dots, m\}, i \neq j} c_A[i]$$

Finpour

Raffiner la base de règles candidates (sélectionner les meilleures règles) « voir algorithme 2 »

Fin algorithme

Algorithme 2. Sélectionner les meilleures règles pour la base de règles

Début algorithme

K = 0 ; stop = false ;

Répéter

R' = argmax{P_R} ;

Si taille base de règles fixées **alors**

Si k < k_{max} **alors**

Ajouter R' dans la base de règle ;

Effacer R' de liste des règles candidates ;

K := k+1 ;

Sinon

Stop = true;

Finsi

Fin algorithme

Algorithme 3. Apprentissage des ensembles flous dans NEFDIAG.

Début d'algorithme

1- sélectionner la forme suivante (p,t) de Γ, propager la, par NEFDIAG et détermine le vecteur de sortie C.

2- pour chaque unité de sortie C_i ; déterminer la valeur de delta

$$\delta C_i = t_i - O_{C_i}$$

3- pour chaque unité de règle R avec O_R > 0

a- déterminer la valeur de δ

$$\delta_R = O_R(1 - O_R) \sum_{C \in U_3} W(R, C) \delta_C .$$

b- trouver x' tel que : $w(x', R)(o_{x'}) = \min_{x \in U_1} \{w(x, R)(\delta_C)\}$

c -pour l'ensemble flou w(x',R) déterminer les valeurs de δ pour les paramètres a,b,c utilisant le pas d'apprentissage σ > 0.

$$\delta_b = \sigma \cdot \delta_R \cdot (c-a) \operatorname{sgn}(O_{x'} - b),$$

$$\delta a = -\sigma \delta_R \cdot (c-a) + \delta_b,$$

$$\delta c = \sigma \delta_R \cdot (c-a) + \delta_b.$$

Appliqué le changement aux w(x',R).

4 –si l'itération est terminée, ou les critères de fin sont validés ; **alors** stop

Sinon aller a (1).

Fin d'algorithme

Algorithme 4. Mises à jour des ensembles floue de conséquents

{ Les paramètres d'entrées suivantes de cet algorithme sont

ν : L'ensemble floue qui doit calculer ses paramètres.

e : Valeur d'erreur.

τ : Degré d'accomplissement pour la règle qui utilise ν dans ça conséquent.

t : Valeur de vecteur de sortie désiré pour le domaine de ν

a, b, c, d Sont les paramètres de ν }

Début algorithme

Si ν est triangulaire **alors**

$$\text{Shift} = \sigma \cdot \tau \cdot (c - a) \cdot \tau \cdot (1 - \nu(t)) ;$$

$$\Delta b = \Delta b \text{ shift} ;$$

Si $\nu(t) > 0$ **alors**

$$\Delta a = \Delta a + \sigma \cdot \tau \cdot (b - a) + \text{shift}$$

$$\Delta c = \Delta c - \sigma \cdot \tau \cdot (b - a) + \text{shift}$$

Sinon

$$\Delta a = \Delta a + \text{sgn}(t - b) \sigma \cdot \tau \cdot (b - a) + \text{shift}$$

$$\Delta c = \Delta c + \text{sgn}(\sigma \cdot \tau \cdot (b - a)) + \text{shift}$$

Finsi

Fin algorithme.

Grammaire pour la spécification en mode texte

Nous pouvons utiliser un fichier pour définir les paramètres de notre système; ensuite NEFDIAG fait une petite compilation pour générer un système neuro-flou utilisant la grammaire de type 0 suivante:

RNF :

```
[<Commentaire>] <Paramètres> [[<commentaire>]<données>] fin
  Commentaire : ligne commence par #
  Paramètres :  PARAMETRES <nouvelle ligne>
                <L1><L2><L3><max_regles><antecbase.nombre tableau>
                <rang de var entrée : tableau de <min max> pairs><nouvelle ligne>
                [UTILISE SOMME<nouvelle ligne>/ UTILISE MAX
                < nouvelle ligne>]
                [INDIVIDUEL <nouvelle ligne>/ MEME < nouvelle ligne>]
                [NOMSVAR<nouvelle ligne><nom_var>]
                <nom_var> :<<chaîne> nouvelle ligne ><chaîne>.....>
  <Données> :  <Ensemble floue>/ <connections>/ <poids>
                <Ensemble floue> : [NOMS <nouvelle ligne>] FLOUE
                <Nouvelle ligne>
                << a b c ls rs [n]><a b c ls rs [n]> .....><nouvelle ligne >
                <Connexion> : MATRICE <nouvelle ligne>
                <Tableau antécédents> <nouvelle ligne>
```

<a b c ls rs[n]> : a= point droite, b :centre, c :point gauche,
 Ls=1 si ensemble de début, rs = 1 si ensemble de la fin

Résumé

L'objectif de cette thèse consiste à faire une implémentation d'un Système d'Inférence Flou de type Takagi-Sugeno dans un Réseau de Neurones de type « Elman » pour automatiser le maximum de la surveillance dynamique d'un système de production. L'atelier de clinkérisation de la SCIMAT, cimenterie d'Ain Touta « Batna » est notre champ d'application privilégié. Afin d'atteindre cet objectif la thèse a été ainsi organisée.

La première partie présente les principales différences entre une surveillance classique et dynamique ainsi que le diagnostic prédictif, le pronostic et enfin la maintenance préventive. La deuxième partie est dédiée aux principales architectures et principes de fonctionnement des systèmes Neuro-Flous dynamique et leurs applications. Dans la troisième partie nous proposons un système Neuro-Flous dynamique pour la surveillance du système de production évolutive.

Abstract

The objective of This thesis is to an implementation of a fuzzy inference system of Takagi-Sugeno neural network in a type "Elman" to automate as much of the dynamic monitoring of a production system that is in our case the sintering workshop SCIMAT, cement Ain Touta "Batna" . To achieve this goal we organize this thesis as well.

The first part presents the main differences between a conventional and dynamic monitoring and predictive diagnosis, prognosis and finally corrective maintenance. The second part is devoted to the main architectures and operating principles of Neuro-Fuzzy systems and their dynamic applications. In the third part, we propose a Neuro-Fuzzy dynamic monitoring of the production system scalable.

ملخص

الهدف من هذه الأطروحة هو تنفيذ نظام الاستدلال الغامض من نوع تاكيجي سيجونو داخل نظام الشبكة العصبية الخاصة ب أيلمان و ذلك من أجل تألية أكبر قدر من المراقبة الدينامكية لنظام الإنتاج و الذي هو في حالتنا ورشة عمل مصنع الاسمنت بعين التوتة باتنة. و لتحقيق هذه الأهداف نقوم بتنظيم هذه الأطروحة كما يلي الجزء الأول يعرض الاختلافات بين المراقبة التقليدية و الدينامكية كذلك التشخيص التنبؤي. و الصيانة التصحيحية . و يخصص الجزء الثاني لأبنية و المبادئ الأساسية للأنظمة العصبية الضبابية و تطبيقاتها. وفي الجزء الثالث نقتراح المراقبة الدينامكية بواسطة الأنظمة العصبية الضبابية الدينامكية للنظام المدروس.

Mots clés:

Diagnostic, réseaux de neurones artificiels, logique floue, systèmes Neuro-Flous, reconnaissance de formes, systèmes Neuro-Flous temporels, RNFE, RdF dynamique, surveillance dynamique, pronostic, maintenance proactive.