



# Pronostic industriel : étude de l'erreur de prédiction du système ANFIS

OTILIA DRAGOMIR<sup>1</sup>, RAFAEL GOURIVEAU<sup>2</sup>, NOUREDDINE ZERHOUNI<sup>2</sup>

<sup>1</sup> Université Valahia de Târgoviste, Faculté d'Ingénierie Electrique  
Département d'Automatique et Informatique  
18-20 Unirii Boulevard, Târgoviste, Roumanie

<sup>2</sup> Institut FEMTO-ST, UMR CNRS 6174 - UFC / ENSMM / UTBM  
Département Automatique et Systèmes Micro-Mécatronique (AS2M)  
24 rue Alain Savary, 25000 Besançon, France

drq\_otilia@yahoo.com, rafael.gouriveau@ens2m.fr, noureddine.zerhouni@ens2m.fr

**Résumé** – Le travail porte globalement sur le développement d'un outil de pronostic de défaillances basé sur l'utilisation d'un système de prédiction neuro-flou. Plus particulièrement, cet article vise la proposition d'une architecture de prédiction basée sur l'utilisation du système ANFIS (système d'inférence floue paramétré par apprentissage neuronal), et pour laquelle différents axes d'améliorations des prédictions sont proposés. La stabilité des erreurs de prédictions en fonction de l'horizon de prédiction est étudiée expérimentalement et une solution visant à intégrer les sollicitations "futures" connues dans le modèle prédictif est proposée. L'ensemble est illustré sur un benchmark de prédiction : la série de données dite de Box-Jenkins.

**Mots-clés** – Pronostic, système neuro-flou, ANFIS, erreur de prédiction, stabilisation

## I. INTRODUCTION

Le maintien en condition opérationnelle des systèmes industriels à moindre coût est devenu un facteur critique quant à la performance des entreprises, et l'activité de "pronostic de défaillances" est aujourd'hui considérée comme un processus clef dans les stratégies de maintenance : plutôt que de comprendre a posteriori un phénomène qui vient de se manifester (défaillance...), il est opportun d'en "anticiper" l'apparition afin de recourir à des actions protectrices en conséquence.

Dans la pratique, l'utilité d'un système de pronostic est intimement liée à sa faculté à approximer et à prédire la dégradation d'un équipement. Aussi, beaucoup de méthodes de prévision peuvent être envisagées pour supporter cette activité de prédiction [1]. La majorité d'entre elles portent sur la construction de modèles capables de minimiser l'erreur de prédiction pour un horizon donné. Cependant, aucune ne traite du contrôle de la distribution de cette erreur en fonction de l'horizon de prédiction. Or, cette problématique nous semble être essentielle en regard du processus global du pronostic : les décideurs, certes nécessitent des indicateurs quant à l'évolution dans le temps de leurs équipements, mais doivent pouvoir y associer une confiance. C'est ce problème de stabilité de l'erreur de prédiction qui est discuté dans ce papier.

L'implémentation d'un outil de prédiction doit s'accommoder des spécificités des systèmes industriels surveillés... Les installations réelles étant complexes et de comportement généralement non stationnaire et non linéaire, une étape de modélisation de leur comportement peut être difficile voire impossible. Le travail rapporté se place dans ce cadre pour lequel peu de connaissance sur le comportement des systèmes est disponible et, par suite et conformément à la littérature, les

travaux sont orientés vers l'exploitation des réseaux adaptatifs, plus spécifiquement, du système neuro-flou ANFIS : différents travaux montrent que le système ANFIS est à même de "capturer" le comportement d'un système rapidement et précisément à court terme, et cela, mieux encore que d'autres méthodes, réseaux de neurones (RNs) inclus. Ceci étant, dans la pratique, la construction d'un système ANFIS se heurte à des problèmes similaires à ceux rencontrés pour les RNs : l'exactitude des prédictions réalisées dépend de beaucoup de facteurs (choix des entrées, nombre et type des fonctions d'appartenance, algorithme d'apprentissage).

Ainsi, l'objet de ce papier est de dégager une architecture appropriée du système ANFIS pour supporter l'activité de prédiction du processus de pronostic. Cette étude est menée de manière expérimentale et vise à satisfaire un objectif de contrôle de l'erreur de prédiction du système ANFIS quel que soit l'horizon de prédiction (tel que suggéré ci-avant).

Le papier est structuré en deux parties. La première porte sur la description rapide du système ANFIS, sur l'identification des améliorations potentielles d'un tel système, et sur la formulation du problème de maîtrise de l'erreur de prédiction. Dans la deuxième partie, différentes architectures de prédiction basées sur le système ANFIS sont proposées et testées : la stabilité des erreurs de prédictions en fonction de l'horizon de prédiction est étudiée expérimentalement et une solution visant à intégrer les sollicitations "futures" connues dans le modèle prédictif est proposée. L'ensemble est synthétisé dans une conclusion et quelques pistes de travail complémentaire sont données.

## II. SYSTEME ANFIS ET MAITRISE DE L'ERREUR DE PREDICTION

### A. ANFIS : Adaptive Neuro-Fuzzy Inference System

Le système ANFIS est une classe de réseau adaptatif proposé par [2]. Il peut être vu comme un réseau de neurones non bouclé pour lequel chaque couche est un composant d'un système neuro-flou. A ce titre, le système ANFIS est un « approximateur universel » et a donc été utilisé dans différentes applications de prédictions [3,4] : étant donnée une situation à la date «  $t$  » (entrée), le système ANFIS est capable (après apprentissage) d'en prédire l'évolution à «  $t+prév$  » (sortie).

Considérons la figure 1 pour décrire ce système et brièvement en expliquer les processus d'apprentissage et d'inférence. [3,4] en proposent une vision plus détaillée.

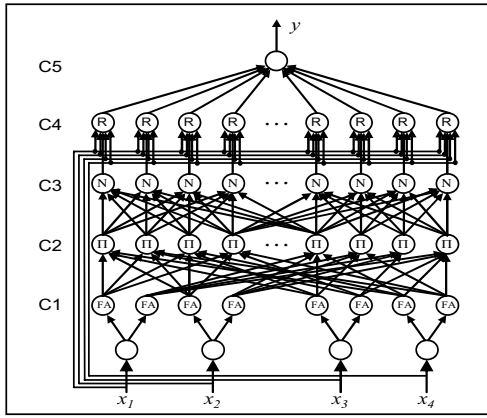


Figure 1. Système ANFIS à 4 entrées et 2 fonctions d'appartenance par entrée

Un système ANFIS réalise une approximation linéaire de la variable de sortie en décomposant l'espace des entrées en différents espaces flous. Exception faite de la couche d'entrée, un système ANFIS est composé de 5 couches. Si deux ensembles flous sont associés à chaque variable d'entrée (figure 1), alors le système présente 16 règles d'inférence  $R_j$  ( $2^4$ ). Ces inférences sont du type Sugeno du premier ordre :

$$R_j : \text{si } (x_1 \text{ est } A_1^j) \text{ et } (x_2 \text{ est } A_2^j) \text{ et } (x_3 \text{ est } A_3^j) \text{ et } (x_4 \text{ est } A_4^j) \\ \text{alors } y_j = c_1^j x_1 + c_2^j x_2 + c_3^j x_3 + c_4^j x_4 + c_5^j \quad (1)$$

Dans la couche 1 (de fuzzification), chaque nœud se comporte comme une fonction d'appartenance (gaussienne, sigmoïde...). Par exemple, des fonctions gaussiennes :

$$\mu_{A_l^j}(x) = \exp(-[x - m_l^j] / b_l^j)^2 \quad (2)$$

avec  $l = \{1, \dots, 4\}$  le numéro de l'entrée et  $\{m_l^j, b_l^j\}$  l'ensemble des paramètres des gaussiennes (paramètres prémisses). Dans la couche 2, chaque nœud réalise une T-norme floue. L'opérateur produit est généralement utilisé.

$$\mu_j = \mu_{A_1^j}(x_1) \mu_{A_2^j}(x_2) \mu_{A_3^j}(x_3) \mu_{A_4^j}(x_4) \quad (3)$$

Dans la couche 3, les sorties des T-normes sont normalisées. Après combinaison linéaire des variables d'entrée dans la couche 4 (paramètres conséquents), la sortie prédite est obtenue dans la couche 5 par une moyenne pondérée des sorties des différentes règles :

$$y = \frac{\sum_{j=1}^{16} \mu_j (c_1^j x_1 + c_2^j x_2 + c_3^j x_3 + c_4^j x_4 + c_5^j)}{\sum_{j=1}^{16} \mu_j} \quad (4)$$

Le système neuro-flou décrit ci-avant comporte 96 paramètres devant être optimisés (16 inhérents aux fonctions gaussiennes et 80 à la linéarisation des sorties de règles) et c'est l'objet de la phase d'apprentissage. Des travaux montrent que la combinaison de l'algorithme de descente du gradient et de l'algorithme l'estimation des moindres carrés permet de réduire

la complexité de l'apprentissage tout en assurant une certaine efficacité [3,5].

### B. Leviers d'action dans la construction d'un système ANFIS

Bien que transparent (modèle boîte grise), un système du type ANFIS doit cependant être construit par l'utilisateur. Aussi, beaucoup de paramètres influencent potentiellement l'erreur de prédiction d'un tel système, et la recherche d'un réseau « performant » peut s'orienter suivant différents axes. Rui [6] a réalisé un inventaire des « facteurs ayant une forte influence sur la précision et l'efficacité d'un réseau de neurones », inventaire qu'il est aisé d'adapter aux réseaux neuro-flous en général, et au système ANFIS en particulier.

1. *Structure du réseau.* Concernant le réseau neuro-flou ANFIS, définir une architecture consiste à choisir le nombre de fonctions d'appartenance, leur type (gaussienne, sigmoïde...) et le nombre de règles floues utilisées. Un compromis simplicité généralisation doit être trouvé : il n'est pas concevable de construire un système ANFIS trop complexe, la base d'apprentissage étant, elle, limitée.
2. *Sélection d'entrées.* La sélection des variables d'entrées est une méthode expérimentale qui vise à déterminer l'ensemble des entrées le plus adéquat pour approximer la relation entrées/sortie. Cette approche est basée sur la construction de groupes d'entrées partiellement disjoints et sur l'évaluation rapide des architectures ainsi obtenues en regard d'un objectif d'apprentissage immédiat (une itération). Dans le même ordre d'idée, différentes approches de partitionnement de l'espace des entrées peuvent être envisagées (grid, tree et scatter portioning) [7].
3. *Base d'apprentissage.* L'ensemble d'apprentissage disponible impacte bien entendu la qualité de généralisation du système ANFIS. Wang [5] rappelle qu'il est nécessaire de disposer d'une base d'apprentissage de taille 5 fois supérieure au nombre de paramètres à optimiser (prémisses et conséquents). Cependant, au-delà de la taille de l'ensemble d'apprentissage, c'est également la pertinence des données qu'il contient qui se révèle critique.
4. *Algorithme d'apprentissage.* [8,9] proposent de modifier l'algorithme classique d'apprentissage afin d'en accélérer la convergence. [10] propose d'optimiser l'initialisation des poids d'apprentissage. La modification des pas d'apprentissage peut aussi être envisagée [11,12]. Le critère d'arrêt des algorithmes d'apprentissage est également difficile à dimensionner afin d'éviter les problèmes de « sur- et de sous-apprentissage » (nombre maximum d'itérations ou seuil de l'erreur admise).

De nombreux paramètres sont donc susceptibles d'influer sur les performances d'un réseau ANFIS. Dans la suite, seuls les points 1 et 2 sont exploités pour tâcher de construire des réseaux permettant de contrôler l'erreur de prédiction. Revenons au préalable sur cette notion de maîtrise de l'erreur.

### C. La maîtrise de l'erreur de prédiction : une mesure de la qualité du pronostic

La qualité d'un système industriel de pronostic dépend grandement de celle de l'approximation de la dégradation de l'équipement. Une grande attention doit donc être portée au modèle prédictif construit à cette fin et deux points peuvent être soulevés [8].

En premier lieu, il est évident que l'erreur de prédiction entre une situation « réelle » et une situation « prédite » doit être le plus faible possible et, au minimum, rester inférieure à un

certain seuil. En d'autres termes, la prédiction réalisée peut (certes) être imprécise (et donc incertaine), mais il doit être possible de prendre des décisions non ambiguës en conséquence. Aussi, afin de pouvoir accorder une certaine confiance au système prédictif, nous jugeons souhaitable d'être à même de contrôler l'erreur de prédiction au détriment, si nécessaire et dans une certaine mesure, de sa valeur. Une illustration de ce propos est proposée en figure 2 – partie haute : bien que non centrée en zéro, la dispersion d'erreur de la courbe 2 laisse présager d'un modèle de prédiction plus fiable que celui de la courbe 1.

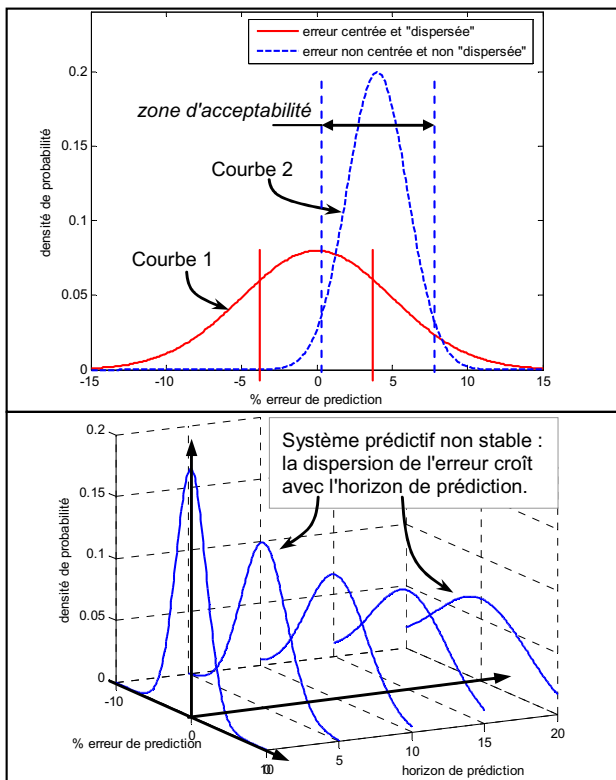


Figure 2. Distributions d'erreurs de prédiction

Une autre problématique, liée à la précédente est celle de la stabilité dans le temps de cette erreur de prédiction. Afin d'assurer une certaine généricité au système prédictif construit, sa « qualité » ne doit être que faiblement dépendante de l'horizon décisionnel pour lequel il est utilisé : les dégradations impactant les performances d'un système industriel sont par essence non déterministes et le modèle prédictif doit pouvoir être exploité à court ou moyen terme (figure 2 - partie basse).

En définitive et compte tenu qu'il est difficile de garantir que minimiser l'erreur de prédiction permet d'en réduire la dispersion, nous proposons de tâcher de satisfaire ce double objectif. Cette maîtrise de l'erreur de prédiction nous semble essentielle dans le cadre du pronostic industriel.

### III. ETUDE EXPERIMENTALE DE L'ERREUR DE PREDICTION MOYEN TERME DU SYSTEME ANFIS

Différentes architectures du système ANFIS ont été testées afin de contrôler l'erreur de prédiction. Nous synthétisons dans cette partie les résultats de cette étude expérimentale.

#### A. Benchmark

Nous avons utilisé la série de données de Box Jenkins pour évaluer l'effet des différentes propositions. Cette série de

données est un benchmark largement utilisé. A l'origine, elle contient 296 couples de données entrées/sortie caractéristiques du taux de gaz carbonique à la sortie d'un fourneau. Pour  $t$  variant de 1 à 296, les ensembles de données décrivent le flux de gaz (combustible) en entrée  $u(t)$ , et la concentration de  $CO_2$  à la sortie  $y(t)$ . A chaque instant  $t$ , les données disponibles sont les suivantes :  $\{u(t-5), u(t-4), u(t-3), u(t-2), u(t-1), u(t), y(t-3), y(t-2), y(t-1), y(t), y(t+1)\}$ . Sur cette base, différents systèmes ANFIS peuvent être envisagés pour réaliser des prédictions de la sortie à tout horizon temporel «  $t+prév$  ».

#### B. Modèles de simulation

Quatre architectures distinctes ont été envisagées. Par simplicité, nous les noterons « modèles » (figure 3).

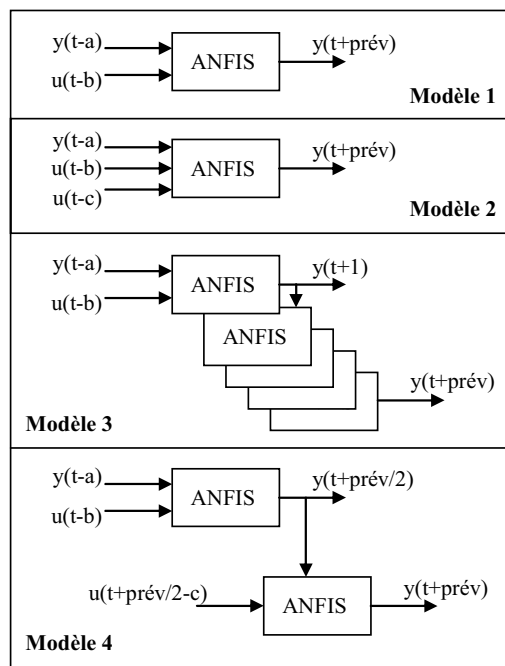


Figure 3. Modèles de simulation

La première architecture correspond au système ANFIS classique à 2 entrées :  $\{y(t-a), u(t-b)\}$ .

La deuxième architecture correspond au système ANFIS classique à 3 entrées :  $\{y(t-a), u(t-b), u(t-c)\}$ , où  $a=0...3$ , et  $b, c=0...5$ .

Le troisième modèle est constitué de la mise en cascade de plusieurs ANFIS à 2 entrées paramétrés pour réaliser des prévisions à «  $t+1$  ». Les prédictions du premier deviennent les entrées du second et ainsi de suite. En terme de complexité, un seul apprentissage est cependant nécessaire.

Le dernier modèle est la mise en série de deux ANFIS à deux entrées, le second devant apprendre les erreurs réalisées par le premier (2 apprentissages distincts). Pour cette architecture, la prédiction est réalisée en considérant le futur programmé connu comme une entrée utile au processus de prédiction. Dans le cadre du pronostic, il est en effet utile de tenir compte de futures modifications du profil de mission telles que des interventions de maintenance programmées. Une illustration est proposée en figure 4. Les entrées du premier ANFIS sont  $\{y(t-a), u(t-b)\}$  et il vise la prédiction de la sortie à la moitié de l'horizon. Les entrées du second ANFIS sont la sortie du premier  $y(t+prév/2)$  ainsi que les interventions programmées sous la forme de  $y(t+prév/2-c)$ . L'ensemble permet de prédire la sortie  $y(t+prév)$ .

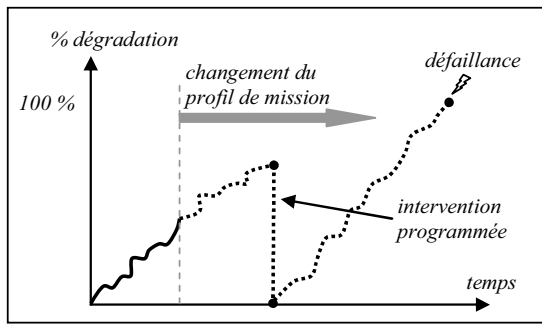


Figure 4. Changement de profil de mission

### C. Conditions de test

En vue de réaliser un jeu de test permettant de comparer les performances des modèles, différents paramètres doivent être fixés au préalable. L'ensemble des conditions de tests pour les prédictions réalisées à «  $t+2$  », «  $t+4$  », «  $t+10$  », «  $t+20$  », «  $t+40$  », «  $t+50$  » est résumé dans le tableau 1.

Le choix d'une mesure d'erreur pour comparer l'efficacité de méthodes de prédiction a été beaucoup discuté dans la littérature et il n'existe pas de consensus global [1]. Pour juger des résultats de simulation, nous avons considéré les indicateurs suivants :

- d'abord, l'erreur quadratique moyenne (notée ici RMSE – root mean square error), qui est le plus commun des indicateurs d'erreur,
- ensuite l'erreur moyenne absolue exprimée en pourcentage (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 donc être complétés par l'analyse de la fonction de densité de probabilité des erreurs sur chacun des tests.

Système d'inférence flou	
Nombre d'entrées	2 ou 3
Nombre de fonctions d'appartenance	2 ou 3
Type de fonction d'appartenance	Gaussienne ou Gauss Bell
T- norme	Produit
Type d'inférence	Linéaire Sugeno
Apprentissage	
Nombre de donnée	290
Nombre d'époques	100
Ensemble d'apprentissage / test	145 / 145
Algorithme d'apprentissage	(LES+ Descente Gradient)

Tableau 1. Les conditions de test

### D. Sélection des entrées

Compte tenu que la série de Box Jenkins est composée d'ensembles de 10 données (hormis la sortie), une méthode doit être employée afin de réaliser la sélection des entrées la plus pertinente possible sur chacun des modèles. Pour cela, nous avons utilisé la méthode proposée par [3] qui consiste à lancer l'apprentissage de différents modèles construits avec différentes entrées (paramètres a, b et c de la figure 3) pour une seule itération de calcul (une seule passe de l'algorithme des moindres carrés). [3] montre que le modèle ayant le plus faible

écart quadratique à l'issue de ce mini-apprentissage est le plus susceptible de réaliser de bonnes prédictions.

Ainsi, par exemple, pour l'identification des entrées les plus pertinentes pour le modèle 2,  $C_3^{10} = 120$  systèmes ANFIS doivent être construits, et une itération d'apprentissage doit être réalisée sur chacun de ces 120 modèles. Ceci reste cependant moins complexe et moins consommateur en temps de calcul que le fait de faire 120 itérations d'apprentissage sur un seul modèle.

A un autre niveau, remarquons que les données de la série de Box Jenkins peuvent être regroupées en deux ensembles, le premier correspondant aux entrées du fourneau  $\{u(t-5), u(t-4), u(t-3), u(t-2), u(t-1), u(t)\}$ , et le second à l'historique des sorties du modèle  $\{y(t-3), y(t-2), y(t-1), y(t)\}$ . De fait, les combinaisons d'entrées possibles sont drastiquement réduites. Pour le modèle 1, seuls  $6 \times 4 = 24$  modèles ANFIS doivent être testés pour réaliser la sélection des entrées. Par exemple, une sélection d'entrées pour ce modèle peut être :  $\{u(t-4), y(t-1)\}$ .

### E. Résultats

*Modèles 1 et 2.* Les courbes de distribution de l'erreur de prédiction des modèles 1 et 2 à «  $t+1$  », «  $t+10$  » et «  $t+20$  » sont proposées en figures 5 et 6. Ces modèles réalisent de bonnes prédictions à court terme mais leurs performances se dégradent rapidement à mesure que l'horizon de prédiction augmente. Ces modèles ne permettent ainsi pas d'obtenir un compromis entre la précision à court terme et la stabilité moyen terme, et s'avèrent donc d'un intérêt limité dans le cadre du pronostic industriel.

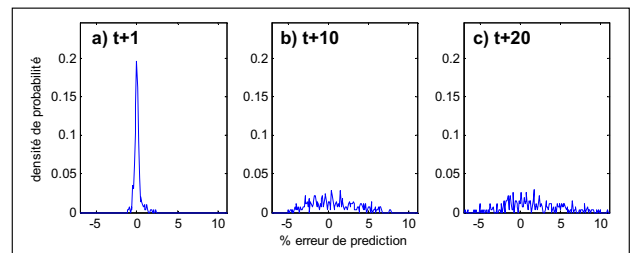


Figure 5. Modèle 1 - distribution de l'erreur de prédiction pour a)  $prév=1$ , b)  $prév=10$ , c)  $prév=20$

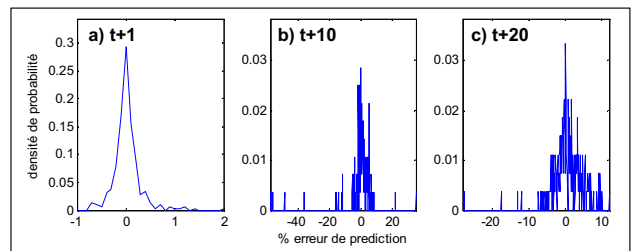


Figure 6. Modèle 2 - distribution de l'erreur de prédiction pour a)  $prév=1$ , b)  $prév=10$ , c)  $prév=20$

*Modèle 3.* Une première manière de stabiliser l'erreur de prédiction est de construire une architecture composée de systèmes ANFIS "en cascade". Nous avons appliqué cette approche dans la conception du troisième modèle (cf. figure 3) : un système ANFIS a été paramétré pour réaliser des prédictions à «  $t+1$  » et a été utilisé "prév" fois afin de réaliser une prédiction à l'horizon «  $t+prév$  ». Pour cela, les sorties estimées à la première itération ( $y(t+1)$ ) deviennent les entrées de la seconde prédiction et ainsi de suite. L'intérêt de

construire ce type de modèle peut être évalué grâce aux indicateurs RMSE et MAPE (figure 7) : l'erreur de prédiction semble se stabiliser pour un horizon à moyen terme, ce qui se reflète sur les courbes de la figure 8.

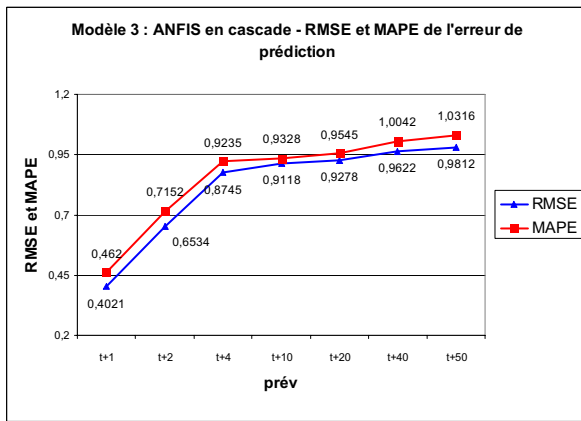


Figure 7. Modèle 3 – RMSE et MAPE de l'erreur de prédiction pour un horizon variant de « t+1 » à « t+50 »

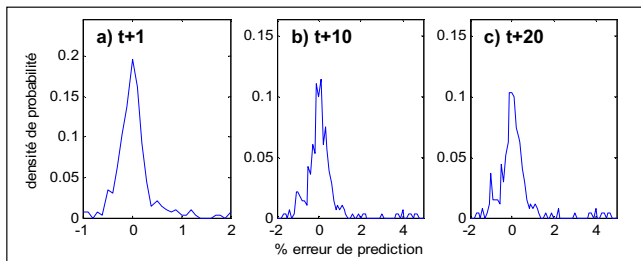


Figure 8. Modèle 3 - distribution de l'erreur de prédiction pour a) prév=1, b) prév=10, c) prév=20

**Modèle 4.** Le quatrième modèle est construit sur la base de deux systèmes ANFIS en série, ce qui permet d'injecter les sollicitations futures connues du système (profil de mission) dans le modèle. Le premier ANFIS a pour entrées  $\{u(t-a), y(t-b)\}$  sélectionnées à l'aide de l'algorithme proposé par [3] et réalise une prédiction à  $(t+prev/2)$ . Le second ANFIS a pour entrées les sorties du premier et les sollicitations prévues, et réalise une prédiction à un horizon  $prev/2$ , ce qui permet au final d'obtenir la sortie  $y(t+prev)$ .

Les distributions des erreurs de prédiction pour des horizons prévisionnels de « t+1, t+10 et t+20 » sont proposées en figure 9. Les résultats des simulations montrent que l'exactitude des prédictions tend à ne pas trop se dégrader : la moyenne et l'écart type des mesures de distribution d'erreur semblent se stabiliser à moyen terme.

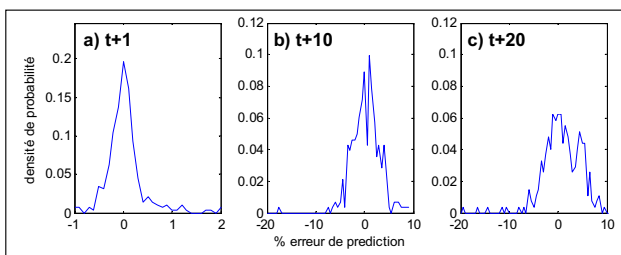


Figure 9. Modèle 4 - distribution de l'erreur de prédiction pour a) prév=1, b) prév=10, c) prév=20

Il est en revanche plus difficile de conclure que ce modèle est préférable au modèle 3. L'utilisation du benchmark de Box Jenkins reste effectivement insuffisante pour bien illustrer l'intérêt du modèle 4 : l'injection des sollicitations futures connues dans le modèle prédictif est d'autant plus pertinente que ces sollicitations sont différentes des sollicitations "classiques", ce qui n'est pas le cas de la série de données de Box Jenkins. Revenant à la problématique du pronostic, il est utile de pouvoir disposer d'une architecture du type de celle du modèle 4 pour le système prédictif.

#### IV. CONCLUSION

Bien que l'activité de pronostic ne devienne un élément clef dans la recherche de stratégies de maintenance industrielle, il existe encore peu de systèmes réels permettant de supporter ce processus. Dans ce cadre, le travail rapporté ici vise la spécification d'un système de pronostic capable de prendre en compte la dynamique des équipements réels.

L'utilité d'un système de pronostic est liée à sa faculté à approximer et à prédire l'évolution de la dégradation d'un équipement. Aussi, beaucoup de méthodes de prévision sont potentiellement utilisables. Deux aspects peuvent alors être soulignés.

1) Le système prédictif choisi doit être à même de s'accommoder des spécificités des systèmes industriels surveillés (complexité, dynamique, non-linéarité, peu de connaissance disponible...). De ce point de vue, les développements partiellement rapportés dans le papier sont centrés sur l'exploitation de systèmes neuro-flous, plus spécifiquement du système ANFIS, qui est à même de capturer le comportement d'un système rapidement et correctement. La qualité des prédictions réalisées grâce à ce type d'outil dépend cependant de l'architecture du système de prédiction choisie...

2) La majorité des développements sur les outils de prédiction portent sur la construction de modèles capables de minimiser l'erreur de prédiction pour un horizon donné. En revanche, aucun ne traite de la stabilisation de la distribution de cette erreur en fonction de l'horizon de prédiction. C'est ce point qui est davantage discuté dans le papier : différentes structures de systèmes ANFIS sont proposées et la stabilité des prédictions résultantes de ces propositions est étudiée expérimentalement, sur un benchmark de prédiction : la série de Box Jenkins.

Deux types architectures du système ANFIS permettent de stabiliser l'erreur de prédiction à moyen terme. La première est l'association série de différents systèmes ANFIS optimisés pour réaliser des prédictions à court terme : les prédictions du premier système deviennent les entrées du second et ainsi de suite. La seconde architecture consiste à construire un modèle composé de deux systèmes ANFIS en cascade réalisant chacun des prédictions à la moitié de l'horizon final. Cette structure permet d'injecter en entrée du deuxième ANFIS les sollicitations futures connues du système surveillé, et de fait, de rendre la qualité (stabilité) du modèle prédictif un peu moins dépendante de l'horizon de prédiction final (à moyen terme).

L'étude est actuellement poursuivie afin de considérer la maîtrise de l'erreur de prédiction comme un objectif explicite lors de la phase d'apprentissage des paramètres du réseau neuro-flou. L'idée sous-jacente est de chercher à trouver un compromis entre généralisation et approximation, en cohérence avec les exigences de précision à court et moyen termes exprimées par l'utilisateur.

## V. REFERENCES

- [1] J.G. De Gooijer, R.J. Hyndman, « 25 years of time series forecasting », *International Journal of Forecasting*, vol. 22, pp. 443-473, 2006.
- [2] J.-S.R. Jang, « ANFIS: adaptive-network-based fuzzy inference systems », *IEEE Transaction on Systems, Man, Cybern.*, vol. 23(3), pp. 665-685, 1993.
- [3] J.S.R. Jang, C.T. Sun, « Neuro-Fuzzy modeling and control », *Proceedings of IEEE*, vol. 83-3, pp. 378-406, 1995.
- [4] R.C.M. Yam, P.W. Tse, L. Li, P. Tu, « Intelligent predictive decision support system for condition-based maintenance », *Int. Journal of Advanced Manufacturing Technology*, vol. 17, pp. 383-39, 2001.
- [5] W.Q. Wang, M.F. Goldnaraghi, F. Ismail, « Prognosis of machine health condition using neuro-fuzzy systems », *Mechanical Systems and Signal Processing*, vol. 18, pp. 813-831, 2004.
- [6] Y. Rui, A.A. El-Keib, « A Review of ANN-based Short-Term Load Forecasting Models », *Proceedings of the 27<sup>th</sup> South-eastern Symposium on System Theory*, Starkville, USA, vol. 78-82, 1995.
- [7] S.L. Chi, « Selecting input variables for fuzzy models », *Journal of Intelligent and Fuzzy Systems*, vol. 4, pp. 243-256, 1996.
- [8] R. Adeline, R. Gouriveau, N. Zerhouni, « Pronostic de défaillances : maîtrise de l'erreur de prédiction », *MOSIM 2008, 7<sup>ème</sup> Conférence internationale de Modélisation et Simulation*, Paris, 2008.
- [9] M. Riedmiller, M. Braun, « A direct adaptive method for faster backpropagation learning: the RPROP algorithm », *Proceedings of the IEEE International Conference on Neural Networks*, vol. 1, pp. 586-591, 1993.
- [10] J.Y. F. Yam, T.W. S. Chow, « A weight initialization method for improving training speed in feed forward neural network », *Neurocomputing*, vol. 30, pp. 219-232, 2000.
- [11] D. Sha, V.B. Bajie, « An on-line hybrid learning algorithm for multilayer perceptron in identification problems », *Computers & Electrical Engineering*, vol. 28(6), pp. 587-598, 2002.
- [12] S.J. Yoo, J.B. Park, Y.H. Choi, « Indirect adaptive control of nonlinear dynamic systems using self recurrent wavelet neural networks via adaptive learning rates », *Information Sciences*, vol. 177(15), pp.3074-3098, 2007.