

Approche Neuronale de la Représentation Multimodèle de Processus Complexes

NESRINE ELFELLY^{1,2}, JEAN-YVES DIEULOT¹, MOHAMED BENREJEB², PIERRE BORNE¹

¹Laboratoire d'Automatique, Génie Informatique et Signal
Ecole Centrale de Lille, 59651 Villeneuve d'Ascq Cedex, France

²Unité de Recherche L.A.R.A. Automatique
Ecole Nationale d'Ingénieurs de Tunis, Campus Universitaire – BP 37, 1002 Tunis Belvédère, Tunisie

Nesrine.Elfelly@ed.univ-lille1.fr, Jean-Yves.Dieulot@polytech-lille.fr, Pierre.Borne@ec-lille.fr,
Mohamed.Benrejeb@enit.rnu.tn

Résumé — L'approche multimodèle a été récemment développée pour pallier aux difficultés rencontrées lors de la modélisation et de la commande de processus non-linéaires complexes. Malgré son succès dans différents domaines, elle se trouve encore confrontée à plusieurs problèmes tels que la détermination du nombre et des paramètres des modèles de la base ainsi que le choix des méthodes de calcul des validités affectées aux modèles. Dans ce papier, nous proposons une approche neuronale pour la représentation multimodèle de processus complexes. La mise en œuvre de cette approche passe par quatre étapes. La première consiste à mettre en œuvre des techniques expérimentales pour récupérer des mesures de type entrée/sortie permettant la description de toutes les dynamiques de fonctionnement du système. Ces mesures vont ensuite servir pour déterminer le nombre de modèles de la base en utilisant un apprentissage compétitif pénalisant le rival. Ensuite, l'utilisation d'une méthode de classification adéquate (cartes de Kohonen ou k-moyennes floues) permet de générer les domaines de fonctionnement du système à partir desquels les modèles sont identifiés. La dernière étape consiste à valider la base de modèles ainsi construite en mettant en œuvre la méthode de calculs des validités la plus convenable. L'approche proposée a été implémentée et testée pour différents types de systèmes. Les résultats obtenus témoignent de la performance et la précision de cette approche.

Mots clés — processus complexes, modélisation, approche multimodèle, méthodes d'apprentissage neuro-floues, validités.

I. INTRODUCTION

Le but principal de l'étude des systèmes est de pouvoir les commander afin d'assurer un comportement désiré. Or, l'établissement d'une commande nécessite la connaissance d'un ou de plusieurs modèles qui doivent représenter au mieux l'évolution du processus étudié.

La complexité d'un processus réel peut se présenter sous différents aspects tels qu'une forte non-linéarité, une non-stationnarité, un vaste domaine de fonctionnement, des changements dans les paramètres du système ou bien aussi des perturbations externes. Par conséquent, il est souvent difficile voire même impossible de concevoir un modèle représentatif global. Une possibilité pratique consiste à recourir à une combinaison d'approches locales en introduisant la notion de représentation multimodèle. Cette approche a été proposée comme une alternative pour appréhender le comportement non linéaire et complexe d'un système par un ensemble de modèles plus simples linéaires ou non, formant ce qu'on appelle base de modèles. Chacun de ces modèles représente le processus dans un domaine particulier de son espace de fonctionnement.

Plusieurs versions de la représentation et de la commande multimodèle [10] ont été envisagées dans différents contextes. En effet, en 1985, Takagi & Sugeno [20] ont proposé une représentation floue du processus par une combinaison de modèles linéaires locaux. Cette approche a connu un grand succès et a été appliquée dans plusieurs domaines mais elle est souvent confrontée à des difficultés de manque d'information sur le processus étudié ou une incertitude paramétrique importante, comme elle peut aboutir parfois à un grand nombre de modèles alourdissant ainsi la tâche de commande. Ceci a incité un grand nombre de chercheurs à développer d'autres approches afin de remédier à ces difficultés.

Ainsi, des approches de détermination de la base de modèles ont été développées pour les systèmes complexes incertains à paramètres bornés, [12,14]. Les approches citées précédemment requièrent la connaissance des limites des variations des paramètres du système, ce qui n'est pas toujours possible notamment pour les systèmes incertains, pour lesquels les domaines de variations des paramètres sont souvent inconnus ou mal connus. D'autres études [13,18] proposent de décrire le système par un ensemble de modèles locaux définis en général à partir d'un modèle de référence linéarisé en plusieurs points.

Par contre, lorsque l'on dispose uniquement de données de type entrée/sortie, il est nécessaire de construire la base de modèles et les modèles même en faisant appel à des techniques d'identification et de classification. Parmi l'éventail des techniques disponibles, nous avons choisi de nous intéresser à des méthodes neuro-floues permettant la détection de la base de modèles. Peu de techniques neuronales ont été appliquées à l'établissement d'une représentation d'un système ou d'un modèle non linéaire par une combinaison variable de plusieurs sous-modèles linéaires ou non [1,5,9,15,16,17,19] mais la plupart des méthodes proposées ne tiennent pas compte des fortes non-linéarités de certains processus ou sont difficiles à mettre en œuvre.

Dans ce papier, nous proposons une approche pratique pour la représentation multimodèle de processus complexes qui ne nécessite ni la définition d'un modèle global ni la connaissance des domaines de variation des paramètres du système, les seules mesures des variables d'entrée/sortie suffisent pour sa mise en œuvre. Elle permet aussi de résoudre le problème de la détermination du nombre et des paramètres des modèles de la base en faisant appel à quelques méthodes de classification neuronales et floues.

Dans la section suivante, nous détaillerons la méthodologie utilisée pour la mise en œuvre de l'approche envisagée. Ensuite, nous présenterons deux exemples de simulation et

quelques interprétations. Nous finirons le présent papier par une conclusion.

II. DESCRIPTION DE L'APPROCHE ENVISAGÉE POUR LA MODELISATION DE PROCESSUS COMPLEXES

L'approche proposée permet la génération automatique du nombre, de la structure et des paramètres des différents modèles de la base. La mise en place de cette approche nécessite tout d'abord la détermination du nombre de modèles en se servant d'un réseau de neurones à apprentissage compétitif pénalisant le rival. L'étape suivante consiste à répartir les données par classes en utilisant une méthode de classification parmi celles qui ont été retenues : cartes de Kohonen et k-moyennes floues. Suivant certains aspects du processus, on fera appel à la méthode la plus convenable. Les résultats de la classification seront par la suite exploités pour mettre en œuvre une identification paramétrique des modèles de la base. La dernière étape consiste à valider la modélisation ainsi élaborée en adoptant une méthode adéquate de calcul des validités pour la génération de la sortie multimodèle qui sera comparée à la sortie réelle du système pour différentes entrées.

A. Détermination du nombre de modèles de la base

Cette étape consiste tout d'abord à exciter le système par une séquence d'entrée appropriée et récupérer les mesures de type entrée/sortie. Ces mesures vont servir pour la détermination du nombre de modèles représentatifs du système en faisant appel à une classification par apprentissage compétitif pénalisant le rival.

1. Excitation du système et récupération des données

L'excitation du système consiste à appliquer un signal d'entrée et ensuite collecter les mesures nécessaires (de sortie ou d'entrée/sortie). Suivant la complexité du système, il faut augmenter ou diminuer le nombre de variables mesurables. Le signal d'excitation doit être assez riche avec des paramètres bien choisis permettant une excitation de toutes les dynamiques de fonctionnement et une prise en compte de l'aspect complexe et non linéaire du processus considéré.

2. Apprentissage compétitif pénalisant le rival pour le calcul du nombre de modèles de la base

Une étape importante et déterminante consiste à déterminer le nombre de classes qui correspond au nombre de modèles de la base. Or, pour la plupart des méthodes de classification par apprentissage non supervisé, le nombre de classes doit être connu a priori. L'apprentissage compétitif pénalisant le rival permet de résoudre ce problème.

En effet, l'apprentissage compétitif pénalisant le rival est une stratégie d'apprentissage non supervisé (proposé par Xu en 1993 [22] et révisée par Tambe en 1996 [21]) qui génère automatiquement le nombre optimal de classes. Le principe soulignant cet apprentissage est similaire à celui de l'apprentissage compétitif (utilisé avec les cartes de Kohonen (section B.1)) sauf qu'à la présentation d'une observation $x(t)$, non seulement le vecteur poids w_g du neurone gagnant g est déplacé dans le sens de l'observation mais aussi son rival w_r (deuxième gagnant r) est repoussé ou pénalisé. Le coefficient $\gamma(t)$ suivant lequel le rival est pénalisé doit être très inférieur au coefficient d'apprentissage $\eta(t)$.

$$g = \arg \min_k \|x(t) - w_k(t)\| ; \quad (1)$$

$$r = \arg \min_{k \neq g} \|x(t) - w_k(t)\| ; \quad (2)$$

$$\begin{cases} w_g(t+1) = w_g(t) + \eta(t)(x(t) - w_g(t)); \\ w_r(t+1) = w_r(t) + \gamma(t)(x(t) - w_r(t)); \\ w_k(t+1) = w_k(t) \quad \forall k \notin \{g, r\}; \end{cases} \quad (3)$$

$$\gamma(t) \ll \eta(t).$$

B. Détermination des classes de fonctionnement du système

Une fois le nombre de modèles déterminé, l'étape suivante consiste à répartir les mesures par classe afin d'obtenir les différents domaines de fonctionnement à partir desquels les modèles de la base sont identifiés. Pour cela, on a retenu deux méthodes de classification par apprentissage non supervisé : les cartes de Kohonen et l'algorithme des k-moyennes floues.

1. Classification par les cartes de Kohonen

La carte auto-adaptative de Kohonen est un réseau de neurones qui est en général à deux dimensions, chaque neurone de la couche d'entrée étant relié à chaque neurone de la couche de sortie. La spécificité de ce type de réseau est l'établissement d'une carte discrète, ordonnée topologiquement, en fonction du patron d'entrée [11]. La stratégie de classification consiste à appliquer la règle de Kohonen, qui est caractérisée par un apprentissage compétitif non supervisé où une compétition a lieu avant la modification des poids du réseau. Seuls les poids des neurones qui gagnent la compétition sont changés.

La règle de Kohonen se présente comme suit :

- les poids sont initialisés à des valeurs aléatoires;
- un vecteur d'entrée $x(t)$ est présenté au réseau;
- la distance entre le vecteur d'entrée et les poids $w_k(t)$ reliant les entrées à chaque neurone de sortie est calculée;
- le neurone g correspondant à la plus petite distance et donc le plus proche du vecteur d'entrée gagne la compétition. Les poids $w_g(t)$ reliant les entrées à ce neurone sont alors actualisés [4].

$$g = \arg \min_k \|x(t) - w_k(t)\| ; \quad (4)$$

$$\begin{cases} w_g(t+1) = w_g(t) + \eta(x(t) - w_g(t)); \\ w_k(t+1) = w_k(t) \quad \forall k \neq g; \end{cases} \quad (5)$$

η : coefficient d'apprentissage.

Cette procédure est répétée plusieurs fois jusqu'à stabilisation des poids.

Pour la classification des mesures issues du système, on utilise un réseau de Kohonen pour lequel le nombre de neurones de la couche d'entrée est égal au nombre de variables mesurables du système et le nombre de neurones de la couche de sortie est égal au nombre de classes déjà calculé. A la fin de l'apprentissage, on obtient les vecteurs représentatifs des différentes classes ainsi que leurs centres.

2. Classification par l'algorithme des k-moyennes floues

L'algorithme des k-moyennes floues (développé par Dunn en 1973 [7] et amélioré par Bezdek en 1981 [3]) permet d'obtenir un regroupement des éléments par une approche floue avec un certain degré d'appartenance, où chaque élément peut appartenir à une ou plusieurs classes.

L'algorithme des k-moyennes floues est basé sur l'optimisation de la fonction objective suivante :

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2, \quad 1 \leq m < \infty; \quad (6)$$

où m : nombre réel supérieur à 1;

u_{ij} : degré d'appartenance de x_i à la classe j ;

x_i : $i^{\text{ème}}$ donnée mesurée;

c_j : vecteur centre de la classe j ;

N : nombre d'observations;

C : nombre de classes;

$\|*\|$: une norme quelconque exprimant la similarité entre les données mesurées et le centre de la classe.

L'algorithme se déroule comme suit :

- Initialiser la matrice $U=[u_{ij}]$, $U^{(0)}$
- A l'étape k : calculer les vecteurs centres $C^{(k)}=[c_j]$ avec $U^{(k)}$:

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m \cdot x_i}{\sum_{i=1}^N u_{ij}^m}. \quad (7)$$

- Actualiser $U^{(k)}$, $U^{(k+1)}$:

$$u_{ij} = \frac{1}{\sum_{r=1}^C \left(\frac{\|x_i - c_j\|}{\|x_i - c_r\|} \right)^{\frac{2}{m-1}}}. \quad (8)$$

- Si $\|U^{(k+1)} - U^{(k)}\| < \varepsilon$ (ε est un critère d'arrêt compris entre 0 et 1) alors STOP; sinon retourner à l'étape 2.

Cette procédure converge vers un minimum local de la fonction J_m . Pour aboutir à la classification finale, une donnée mesurée appartient à la classe qui présente le plus grand degré d'appartenance.

Hypothèse :

Les trois techniques précédemment décrites permettent de répartir les mesures issues du système en un nombre fini de classes. Pour un processus donné, on se trouve confronté au problème du choix de la méthode donnant les meilleurs résultats. Dans ce contexte, on a émis une hypothèse qui suppose que le choix dépend de la façon dont les données sont réparties dans l'espace d'observation. En effet, si on remarque qu'il y a un chevauchement entre les données sans séparations nettes, on va privilégier la méthode des k-moyennes floues. Dans le cas contraire, on fera appel à une classification par un réseau de Kohonen. On essaiera de confirmer cette hypothèse à travers une comparaison entre les résultats donnés par ces deux techniques de classification appliquées lors de la modélisation de deux systèmes différents.

C. Identification paramétrique des modèles de la base

A la fin de la classification, on obtient une répartition bien définie de l'ensemble des classes de fonctionnement du système. A chaque classe, correspond un ensemble de mesures de sortie ou d'entrée/sortie. Ces mesures sont exploitées pour l'identification des modèles de la base, qui seront dans notre cas linéaires. Pour ce faire, on procède tout d'abord à une estimation de leurs ordres respectifs. La méthode retenue est celle du test du rapport des déterminants instrumentaux, qui est basée essentiellement sur les conditions de rang de la matrice

d'information contenant les couples de mesures entrée/sortie [2]. Cette matrice est définie comme suit :

$$Q_m = \frac{1}{N_H} \sum_{k=1}^{N_H} \begin{bmatrix} u(k) \\ u(k+1) \\ u(k-1) \\ \vdots \\ u(k-m+1) \\ u(k+m) \end{bmatrix} [y(k+1) \ u(k+1) \ \dots \ y(k+m) \ u(k+m)]; \quad (9)$$

où N_H est le nombre d'observations. Le rapport des déterminants instrumentaux (RDI) est donné par :

$$RDI(m) = \left| \frac{\det(Q_m)}{\det(Q_{m+1})} \right|. \quad (10)$$

Pour chaque valeur de m , la procédure de détermination de l'ordre construit les matrices Q_m et Q_{m+1} et évalue le rapport $RDI(m)$, l'ordre du modèle correspond alors à la valeur de m pour laquelle on observe une augmentation significative du rapport $RDI(m)$.

Connaissant les ordres des différents modèles de la base, la deuxième étape consiste à estimer les paramètres de ces modèles. Pour cela, on a fait appel à la méthode des Moindres Carrés Récursifs (MCR) [2].

D. Calcul des degrés de pertinence et validation de la modélisation proposée

Les trois étapes précédentes sont effectuées hors ligne et ont permis de construire la base de modèles. Notre but est de tester l'efficacité et la précision de la modélisation ainsi proposée. Pour ce faire, on applique au système des entrées différentes de celle qui a été utilisée pour la collecte de mesures. Ensuite, on calcule la sortie multimodèle que l'on compare à la sortie réelle du système. La sortie multimodèle y_{MM} est obtenue par fusion des sorties des modèles y_i pondérés par leurs coefficients de validité respectifs v_i , comme l'illustre la relation suivante :

$$y_{MM}(k) = \sum_{i=1}^N y_i(k) v_i(k); \quad \sum_{i=1}^N v_i(k) = 1. \quad (11)$$

1. Calcul des validités des modèles

Le coefficient de validité est un nombre réel compris entre 0 et 1. Il représente le degré de pertinence de chaque modèle calculé à chaque instant. Dans la littérature, plusieurs méthodes ont été proposées pour le calcul des validités, et nous avons adopté l'approche des résidus. Cette méthode est basée sur la mesure de la distance entre le processus et le modèle considéré. Par exemple, l'expression du résidu peut être définie comme suit :

$$r_i = |y - y_i| \quad i = 1, \dots, N; \quad (12)$$

où N : nombre de modèles de la base;

y : sortie du système;

y_i : sortie du modèle M_i .

Si la valeur du résidu est nulle alors le modèle représente idéalement le système considéré. Dans le cas contraire, le modèle représente partiellement le système. Les résidus normalisés sont donnés par :

$$r'_i = \frac{r_i}{\sum_{i=1}^N r_i}. \quad (13)$$

Dans le contexte de l'approche des résidus, plusieurs méthodes ont été proposées pour le calcul des validités [6,13]. Seulement deux méthodes vont être considérées : les validités simples et les validités renforcées. Les validités sont données par l'expression suivante :

$$v_i = 1 - r_i' \quad (14)$$

A partir de cette expression, nous définissons les validités simples et les validités renforcées.

Validités simples : les validités simples normalisées sont définies de telle sorte que leur somme soit égale à 1 :

$$v_i^{simp} = \frac{v_i}{N-1} \quad (15)$$

Validités renforcées : pour ce type de validités, on utilise la formule de renforcement :

$$v_i^{renf} = v_i \prod_{\substack{j=1 \\ j \neq i}}^N (1 - v_j) \quad (16)$$

Les validités renforcées normalisées s'écrivent alors :

$$v_i^{renf} = \frac{v_i^{renf}}{\sum_{i=1}^N v_i^{renf}} \quad (17)$$

Quant au choix de la méthode de calcul des validités (simples ou renforcées), on a procédé dans [8] à une étude comparative des résultats donnés par les deux méthodes considérées pour différents systèmes. Cette étude a permis de conclure que le choix de la méthode la plus convenable dépend des résultats de la classification. En effet, quand on observe plusieurs variations au sein de la même classe et un chevauchement entre les différentes classes, la méthode des validités simples semble être plus efficace puisqu'elle permet de tenir compte de la contribution des sorties des différents modèles, en se référant à l'expression (15). Dans ce cas, aucun modèle ne peut représenter idéalement le système à aucun instant. Mais lorsque les classes présentent peu de variations et sont bien séparées, la méthode des validités renforcées est mieux adaptée. Grâce à l'expression de renforcement (16), l'application de cette méthode renforce la contribution du modèle le plus dominant qui représente au mieux le comportement du processus.

2. Validation de la modélisation proposée

Ayant choisi la méthode convenable pour le calcul des validités, la validation du modèle global (ensemble de tous les modèles de la base) est effectuée en comparant la sortie multimodèle à la sortie réelle par application de différentes entrées au système.

III. EXEMPLES DE SIMULATION

Afin de souligner l'intérêt et la performance de l'approche décrite dans ce papier, quelques exemples de simulations ont été considérés.

A. Exemple 1 : Système discret du second ordre avec des paramètres variant en fonction du temps

Le système considéré est un système discret complexe dont l'évolution est décrite par l'équation suivante :

$$y(k) = -a_1(k)y(k-1) - a_2(k)y(k-2) + b_1(k)u(k-1) + b_2(k)u(k-2) \quad (18)$$

Les lois de variation des paramètres du processus sont données par :

$$a_1(k) = 0.04 \sin(0.035k) - 0.8, a_2(k) = 0.005 \sin(0.03k) + 0.1 \quad (19)$$

$$b_1(k) = 0.02 \sin(0.03k) + 0.5, b_2(k) = 0.01 \sin(0.035k) + 0.2 \quad (20)$$

Tout d'abord, le système est excité par un signal aléatoire uniforme $u(k)$. On récupère ensuite les mesures de sortie : $y(k)$ et $y(k-1)$. L'ensemble de ces mesures est utilisé pour déterminer le nombre adéquat de classes de fonctionnement en faisant appel à un réseau de neurones à apprentissage compétitif pénalisant le rival. La figure 1 montre qu'avec une couche d'entrée de cinq neurones, deux des centres se sont éloignés de l'ensemble des observations, ce qui donne un nombre de classes égal à trois.

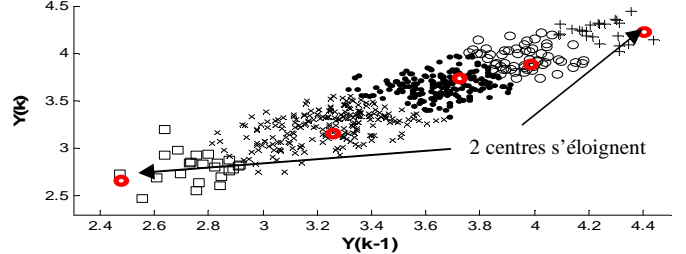


Figure 1 : Détermination du nombre de classes

Ensuite, on procède à une répartition des mesures par classes en mettant en œuvre les deux techniques de classification décrites dans le paragraphe B (réseau de Kohonen et k-moyennes floues). La différence entre ces deux techniques n'est pas suffisamment visible sur les résultats de la classification. En considérant un réseau de Kohonen comme technique de classification, ces résultats sont donnés par la figure 2.

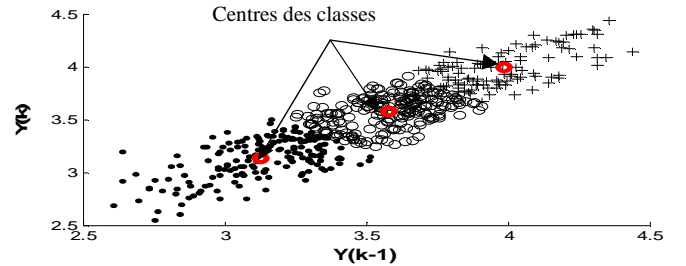


Figure 2 : Résultats de la classification (réseau de Kohonen)

A partir des mesures relatives aux différentes classes obtenues en appliquant chacune des techniques de classification, on estime à chaque fois les ordres et les paramètres des fonctions de transferts relatives aux trois modèles de la base. La figure 3 montre les évolutions des $RDI_i(m)$ pour les trois modèles pour une classification par un réseau de Kohonen.

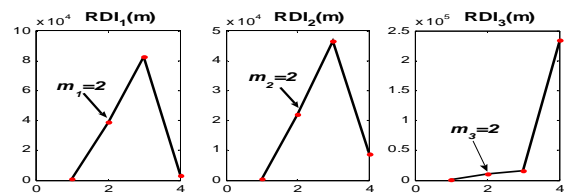


Figure 3 : Evolution des RDI des trois modèles

On remarque que les ordres des modèles sont égaux à deux. Ce résultat est prévisible puisque le système est du second ordre. Ensuite, l'application de la méthode des moindres carrés récursifs nous a permis d'écrire les fonctions de transfert des trois modèles.

Le choix de la méthode de calcul des validités pour ce système a été basé sur une comparaison faite dans [8] qui a montré que les validités renforcées sont mieux adaptées pour ce système. Ceci s'explique par les résultats de la classification qui présentent des classes bien séparées avec peu de variations (figure 2).

Pour la validation de la base de modèles ainsi établie, on se propose d'appliquer la séquence d'entrée suivante :

$$u(k) = 1.5 + 2 \sin(0.06k). \quad (21)$$

On va essayer de comparer les résultats obtenus par application des deux méthodes de classification et par conséquent vérifier l'hypothèse émise dans le paragraphe B. Pour chaque technique de classification, on considère la base de modèles construite, on calcule la sortie multimodèle (fusion des sorties partielles des modèles pondérées par les validités renforcées) et on la compare à la sortie réelle du système. Les résultats sont donnés par la figure 4 où :

y : sortie réelle du système ;

y_{koh} , y_{kmf} : sorties multimodèles obtenues par application des techniques de classification respectives : cartes de Kohonen et k-moyennes floues.

$$e_{koh}(k) = \frac{|y(k) - y_{koh}(k)|}{y(k)}, e_{kmf}(k) = \frac{|y(k) - y_{kmf}(k)|}{y(k)}. \quad (22)$$

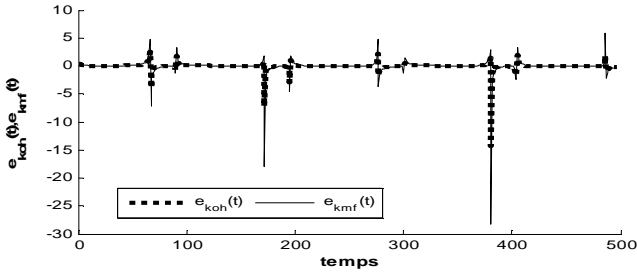


Figure 4 : Erreurs relatives entre la sortie réelle et les sorties multimodèles

On remarque que l'erreur $e_{koh}(t)$ est inférieure à $e_{kmf}(t)$. On constate alors que les meilleurs résultats sont obtenus par application d'une classification par un réseau de Kohonen, ce qui permet de confirmer une partie de l'hypothèse affirmant que dans le cas de données présentant peu de variations, la technique de classification par carte de Kohonen est plus discriminante.

D'autre part, la figure 5 montre que l'application de l'approche proposée permet une bonne modélisation du système. En effet, la sortie multimodèle suit presque parfaitement la sortie réelle pour la séquence d'entrée donnée par l'expression (21).

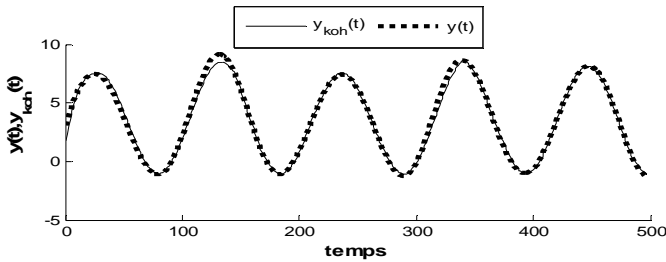


Figure 5 : Sorties réelle et multimodèle (Carte de Kohonen)

B. Exemple 2 : Système continu du second ordre avec des paramètres variant en fonction de l'entrée et de la sortie

Notre deuxième exemple de simulation est un système dont l'évolution est régie par les équations suivantes :

$$a_0 y + a_1 \dot{y} + \ddot{y} = u + b \dot{u}. \quad (23)$$

$$a_0(y) = 0.3 + \text{sigm}(y - 2), a_1(y) = \text{sat}(y^2), b(u) = \text{sat}(u); \quad (24)$$

sigm est la fonction sigmoïde ; sat est la fonction saturation.

Notre deuxième exemple est un système complexe et fortement non linéaire avec des paramètres variant en fonction de l'entrée et de la sortie, ce qui fait de la modélisation une tâche difficile. Le signal retenu pour l'excitation du système est une sinusoïde perturbée par un signal aléatoire, plus riche qu'un simple signal aléatoire et permettant une prise en compte de l'aspect complexe et non linéaire du système. Il est à signaler qu'il a fallu bien ajuster les paramètres du signal (fréquence, amplitude) pour aboutir à de bons résultats.

D'autre part, afin de décrire correctement les dynamiques de fonctionnement du système, on a dû augmenter le nombre de variables mesurables par rapport au premier exemple. En effet, il a fallu considérer non seulement les mesures de sortie $y(k)$ et $y(k-1)$, mais aussi les mesures de l'entrée $u(k-1)$.

La détermination du nombre de classes a été effectuée par un réseau de neurones à apprentissage compétitif pénalisant le rival avec K neurones pour la couche d'entrée et trois neurones pour la couche de sortie. Quelque soit la valeur de K , le nombre de classes obtenu est toujours égal à quatre (figure 6).

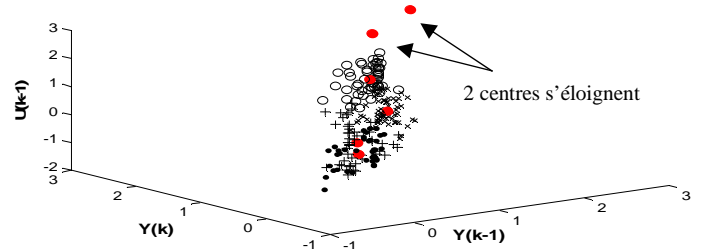


Figure 6 : Détermination du nombre de classes ($K = 6$)

Une fois le nombre de classes déterminé, les mesures sont ensuite réparties en classes en faisant appel aux deux techniques de classification : cartes de Kohonen et k-moyennes floues. Pour le cas de l'algorithme des k-moyennes floues, la répartition des données est donnée par la figure 7.

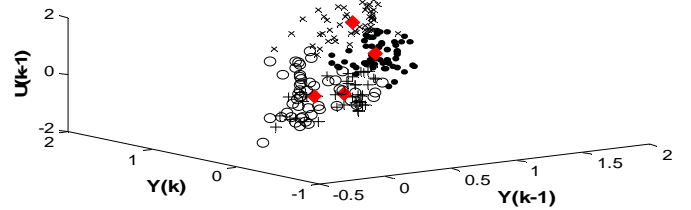


Figure 7 : Résultats de la classification (k-moyennes floues)

Ainsi, la classification aboutit à quatre ensembles de données. Pour chacune des techniques de classification, l'application de la méthode du test du rapport des déterminants instrumentaux pour l'estimation des ordres des modèles et celle des moindres carrés récursifs pour leur identification a permis d'écrire quatre fonctions de transfert ayant des ordres différents.

Pour le choix de la méthode de calcul des validités des modèles, la comparaison élaborée dans [8] a montré que la méthode des validités simples est plus convenable que celle des validités renforcées, ceci est justifié par les résultats de la classification (figure 8) qui présentent un recouvrement entre les classes et plusieurs variations au sein de la même classe.

Pour tester l'efficacité et la précision de la modélisation ainsi proposée, on considère la séquence d'entrée suivante :

$$u(k) = 1.2 + 1.5 \sin(2k). \quad (25)$$

Pour chaque technique de classification, on considère la base de modèles construite et on calcule la sortie multimodèle à partir d'une fusion entre les sorties partielles pondérées par les validités simples. La figure 8 donne les erreurs relatives entre la sortie réelle du système et les sorties multimodèles. On constate qu'à partir d'un certain moment, l'erreur $e_{kmf}(t)$ est presque nulle comparée à $e_{koh}(t)$ qui présente quelques petites variations. On peut alors conclure que, contrairement au premier exemple de simulation, la classification des mesures par l'algorithme des k-moyennes floues est mieux adaptée pour ce système, ce qui permet de confirmer la deuxième partie de notre hypothèse (paragraphe B) qui affirme que lorsqu'on est en présence d'un recouvrement entre les classes,

la technique de classification par l'algorithme des k-moyennes floues est plus convenable.

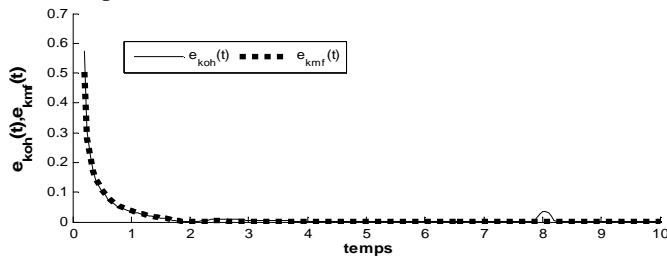


Figure 8 : Erreurs relatives entre la sortie réelle et les sorties multimodèles

La figure 9 justifie la précision et la performance de la modélisation proposée pour le système considéré. En effet, par application de la séquence d'entrée donnée par l'expression (25), les sorties réelle et multimodèle sont presque confondues.

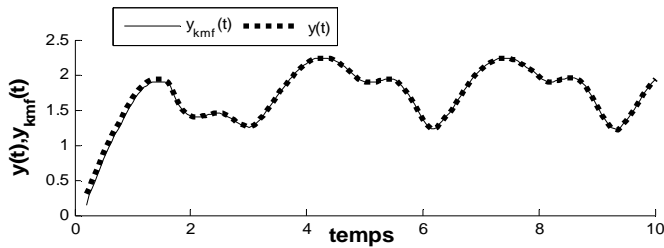


Figure 9 : Sorties réelle et multimodèle (k-moyennes floues)

IV. CONCLUSION

Dans ce papier, nous avons proposé une approche neuronale pratique pour la représentation multimodèle de processus complexes. Cette approche peut être appliquée pour différents types de systèmes (incertains, fortement non linéaires, discrets, continus...). Elle permet la détermination de la base de modèles représentatifs du système dans les différents domaines de fonctionnement par la mise en œuvre de différentes techniques de classification neuro-floues et de deux méthodes d'identification structurelle et paramétrique. Le problème du calcul du nombre de modèles a été résolu par l'utilisation d'un apprentissage compétitif pénalisant le rival. Les différentes étapes de la mise en œuvre de l'approche proposée ont été détaillées. Ces étapes ont été implémentées pour la modélisation de différents systèmes complexes. Les exemples de simulation illustrent l'efficacité et la précision de la stratégie de modélisation proposée, même pour des systèmes fortement non linéaires. Concernant le choix de la méthode de classification des mesures du système (cartes de Kohonen ou k-moyennes floues), une hypothèse a été émise et confirmée à travers les résultats de simulation. En perspectives, il serait intéressant d'étudier la robustesse de l'approche proposée au bruit et aux valeurs aberrantes.

V. REFERENCES

[1] I. S. Baruch, R. B. Lopez, J-L. Olivares, J-M. Flores, « A fuzzy-neural multi-model for nonlinear systems identification and control », *Fuzzy sets and systems*, vol. 159, pp. 2650-2667, 2008.

[2] R. Ben Abdennour, P. Borne, M. Ksouri, F. M'sahli, *Identification et commande numérique des procédés industriels*, Editions Technip, Paris, France, 2001.

[3] J.C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, New York, 1981.

[4] P. Borne, M. Benrejeb, J. Haggège, *Les réseaux de neurones*, Editions Technip, Paris, 2007.

[5] J. Cho, J. C. Principe, D. Erdogmus, M. A. Motter, « Quasi-sliding mode control strategy based on

multiple-linear models », *Neurocomputing*, vol. 70, pp. 960-974, 2007.

[6] F. Delmotte, L. Dubois, P. Borne, « A general scheme for multi-model controller using trust », *Mathematics and Computers in Simulation*, vol. 41, pp. 173-186, 1996.

[7] J. C. Dunn, « A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters », *J. of Cybernetics*, vol. 3, pp. 32-57, 1973.

[8] N. Elfelly, J-Y. Dieulot, P. Borne, « A neural approach of multimodel representation of complex processes », *Int. J. of Computers, Communications & Control*, vol. 3, pp. 149-160, 2008.

[9] Y. Fu, T. Chai, « Nonlinear multivariable adaptive control using multiple models and neural networks », *Automatica*, vol. 43, pp. 1101-1110, 2007.

[10] T. A. Johansen, B. A. Foss, « Editorial: Multiple model approaches to modelling and control », *Int. J. of Control*, vol. 72, pp. 575, 1999.

[11] T. Kohonen, « The self-organizing map », *IEEE Proc.*, vol. 78, pp. 1464 - 1480, 1990.

[12] M. Ksouri-Lahmari, P. Borne, M. Benrejeb, « Multimodel: the construction of Model Bases », *Studies in Informatics and Control*, vol. 3, pp. 199-210, 2004.

[13] D. J. Leith, W. E. Leithead, « Analytic framework for blended multiple model systems using linear local models », *Int. J. of Control*, vol. 72, pp. 605-619, 1999.

[14] S. Mezghani, A. El Kamel, P. Borne, « Multimodel control of discrete systems with uncertainties », *Int. J. of Studies in Informatics and Control*, 2000.

[15] J. C. Principe, L. Wang, M. A. Motter, « Local dynamic modelling with self-organizing maps and application to nonlinear system identification and control », *IEEE Proc.*, vol. 86, pp. 2240-2258, 1998.

[16] M. Ronen, Y. Shabtai, H. Guterman, « Hybrid model building methodology using unsupervised fuzzy clustering and supervised neural networks », *Biotechnology and Bioengineering*, vol. 77, pp. 420-429, 2002.

[17] N. Sadati, A. Talasaz, « Robust fuzzy multimodel control using variable structure system », *IEEE Conf. on Cybernetics and Intelligent Systems*, vol. 1, pp. 497-502, 2004.

[18] R. Schorten, R. Marry-Smith, R. Bjorgan, H. Gollee, « On the interpretation of local models in blended multiple model structures », *Int. J. of Control*, vol. 72, pp. 620-628, 1999.

[19] R. R. Selmic, F. L. Lewis, « Multimodel neural networks identification and failure detection of nonlinear systems », *IEEE Conf. on Decision and Control*, vol. 4, pp. 3128-3133, 2001.

[20] T. Takagi, M. Sugeno, « Fuzzy identification of systems and its applications to modelling and control », *IEEE Trans. on Systems Man and Cybernetics*, vol. 15, pp. 116-132, 1985.

[21] S.S. Tambe, B.D. Kulkarni, P.B. Deshpande, *Elements of artificial neural networks with selected application on chemical engineering, and chemical and biological sciences*, Louisville-KY, 1996.

[22] L. Xu, A. Krzyzak, E. Oja, « Rival penalized competitive learning for cluster analysis RBF and curve detection », *IEEE Trans. on Neural Networks*, vol. 4, pp. 636-649, 1993.