

Revue Internationale de

ISSN 0980-1472

systemique

Vol. 10, N° 5, 1996

afcet

DUNOD

AFSCET

Revue Internationale de
systemique

Revue
Internationale
de Sytémique

volume 10, numéro 5, pages 427 - 446, 1996

Modèles neuroflous pour la modélisation
de systèmes flous

Riad Hartani

Numérisation Afscet, août 2017.



Creative Commons

Rédacteur en chef : B. Paulré
Rédacteur en chef adjoint : E. Andreevsky

Comité scientifique

J. Aracil, Université de Séville; H. Atlan, Université Hébraïque de Jérusalem; A. Bensoussan, Institut National de Recherche en Informatique et en Automatique; M. Bunge, Université McGill; C. Castoriadis, École des Hautes Études en Sciences Sociales; G. Chauvet, Université d'Angers; A. Danzin, Consultant indépendant; P. Davous, EURE-QUIP; J. P. Dupuy, CREA - École Polytechnique; H. Eto, Université de Tsukuba; H. von Foerster, Université d'Illinois; N.C. Hu, Université de Technologie de Shanghai; R. E. Kalman, École Polytechnique Fédérale de Zurich; G. Klir, Université d'État de New York à Binghamton; E. Laszlo, Institution des Nations Unies pour la Formation et la Recherche; J.-L. Le Moigne, Université Aix-Marseille II; J. Lesourne, Conservatoire National des Arts et Métiers; L. Löfgren, Université de Lund; N. Luhmann, Université de Bielefeld; M. Mesarovic, Université Case Western Reserve; E. Morin, École des Hautes Études en Sciences Sociales; E. Nicolau, École Polytechnique de Bucarest; A. Perez, Académie Tchecoslovaque des Sciences; E. W. Ploman, Université des Nations Unies; I. Prigogine, Université Libre de Bruxelles; B. Roy, Université Paris-Dauphine; H. Simon, Université Carnegie-Mellon; L. Sfez, Université Paris-Dauphine; R. Trappl, Université de Vienne; R. Thom, Institut des Hautes Études Scientifiques; F. Varela, CREA - École Polytechnique.

Comité de rédaction

Bureau

D. Andler, CREA - École Polytechnique (*Rubrique Cognition*); E. Andreevsky, Institut National de la Santé et de la Recherche Médicale (Rédacteur en chef adjoint); H. Barreau, Centre National de la Recherche Scientifique (*Rubrique Archives*); E. Bernard-Weil, CNEMATER - Hôpital de la Pitié (*Rubrique Applications*); B. Bouchon-Meunier, Centre National de la Recherche Scientifique (*Rubrique Applications*); P. Livet, CREA - École Polytechnique (*Rubrique Fondements et Épistémologie*); T. Moulin, École Nationale Supérieure des Techniques Avancées (*Rubrique Théorie*); B. Paulré, Université de Paris I, Panthéon-Sorbonne (Rédacteur en chef); J. Richalet, ADERSA (*Rubrique Applications*); R. Vallée, Université Paris-Nord (*Rubrique Théorie*); J.-L. Vullierme, Université de Paris I (*Rubrique Fondements et Épistémologie*).

Autres membres

J.-P. Algoud, Université Lyon-II; A. Dussauchoy, Université Lyon-I; E. Heurgon, Régie Autonome des Transports Parisiens; M. Karsky, ELF-Aquitaine - CNRS; M. Locquin, Commissariat Général de la Langue Française; P. Marchand, Aérospatiale - Université Paris-I; J.-F. Quilici-Pacaud, Chercheur en Technologie; A. Rénier, Laboratoire d'Architecture n° 1 de l'UPA 6; J.-C. Tabary, Université Paris-V; B. Walliser, École Nationale des Ponts et Chaussées; Z. Wolkowski, Université Pierre-et-Marie-Curie.

Membres correspondants

ARGENTINE : C. François (Association Argentinienne de Théorie Générale des Systèmes et de Cybernétique). BELGIQUE : J. Ramaekers (Facultés Universitaires de Notre-Dame de la Paix). BRÉSIL : A. Lopez Pereira (Université Fédérale de Rio de Janeiro). ESPAGNE : R. Rodriguez Delgado (Société Espagnole des Systèmes Généraux). ÉTATS-UNIS : J.-P. Van Gighem (Université d'État de Californie). GRÈCE : M. Decleris (Société Grecque de Systémique). ITALIE : G. Teubner (Institut Universitaire Européen). MAROC : M. Najim (Université de Rabat). MEXIQUE : N. Elohim (Institut Polytechnique National). SUISSE : S. Mumari (Université de Lausanne).

Revue Internationale de Systémique is published 5 times a year: March, May, July, September, December. Date of issue: December 1996.
Second-class postage paid at Rahway, N.J. ISSN N° 0980-1472, USPS N° 007728.
U.S. Mailing Agent: Mercury Air-freight Intl. Ltd., 2323 Randolph Ave., Avenel, NJ07001.
Published by Dunod, 5, rue Laromiguière, 75005 Paris, France and Gauthier-Villars North America Inc., 875-81 Massachusetts Avenue, Cambridge, MA 02139, USA.
Postmaster. Please send all address corrections to: Dunod, c/o Mercury Air-freight Intnk. Ltd. 2323 Randolph Ave., Avenel, NJ 07001, USA.

MODÈLES NEUROFLOUS POUR LA MODÉLISATION DE SYSTÈMES FLOUS

Riad HARTANI ^{1,2}

Résumé

Nous présentons une méthodologie générale pour la modélisation de systèmes flous par modèles neuroflous. Une formalisation des approches neurofloues, dans un contexte d'apprentissage supervisé est réalisée. Nous utilisons les capacités d'apprentissage des réseaux de neurones permettant l'identification de la structure interne du système flou. Les problèmes posés par ce type de modélisation sont discutés et des éléments de réponse apportés.

Abstract

We present a general methodology for fuzzy modeling using neurofuzzy models. We first describe a formalism for neurofuzzy models, in the context of supervised learning. We use neural networks learning capabilities for the identification of the internal structure of fuzzy models. Some of the problems arising from this modeling scheme are discussed and some issues described.

I. INTRODUCTION

Allier les avantages des systèmes flous et des réseaux de neurones sont les principales motivations à l'élaboration d'approches hybrides neurofloues. Plus précisément, le but est d'obtenir des systèmes qui ont à la fois les capacités d'apprentissage, de modélisation et de traitement temps réel des réseaux de neurones ainsi que les capacités de raisonnement et d'interprétation des systèmes flous.

1. INRETS (Institut National de Recherche sur les Transports et leur Sécurité), 2, avenue du Général Malleret-Joinville, 94114 Arcueil, France.
2. LAFORIA-IBP (Laboratoire Formes et Intelligence Artificielle - Institut Blaise-Pascal), Université Paris-VI, boîte 169, 4, place Jussieu, 75252 Paris Cedex 05, France.

Nous présentons en premier lieu une vue générale sur la modélisation de systèmes flous. Celle-ci comporte trois phases. La seconde d'entre elles : l'identification de la structure interne des systèmes flous, nous intéresse particulièrement. C'est à ce niveau que nous analysons l'intégration des techniques neuronales. Nous développons une formalisation des approches neurofloues dans un contexte d'apprentissage supervisé. Différents problèmes spécifiques à ce mode d'apprentissage pour la conception de systèmes flous sont soulevés et discutés. Nous proposons finalement une méthodologie générale pour la conception de modèles neuroflous.

II. MODÉLISATION DE SYSTÈMES FLOUS

Le processus de la modélisation de systèmes flous revient à déterminer une base de règles floues réalisant une fonction de transfert entre un espace d'entrée, représentant l'espace des vecteurs d'état et un espace de sortie, représentant l'espace des vecteurs de commande. La modélisation de systèmes flous constitue un domaine d'investigations très actif (Sugeno and Yasukawa, 1993), (Ralescu and Hartani, 1995). Le processus de modélisation se compose de trois étapes (fig. 1) :

(1) *Identification de la structure externe* : un problème très important est de décider des variables d'entrée et de sortie à utiliser lors de la modélisation. Ce problème est similaire à celui du choix des attributs dans un processus d'apprentissage de concepts. On peut distinguer deux niveaux lors du choix des variables. Le premier consiste à déterminer l'ensemble des variables candidates. Elles doivent être extraites de l'ensemble des variables pouvant caractériser le système. C'est donc un problème d'induction pour lequel aucune solution optimale ne peut être trouvée. Il s'agit donc de faire ce choix de façon heuristique. Le second consiste à réellement définir les variables nécessaires et suffisantes pour la modélisation. Elles forment donc une sous-famille de l'ensemble des variables candidates préalablement défini. Le choix de ces variables est fait de façon à donner au système flou les meilleures performances en description et en généralisation.

(2) *Identification de la structure interne* : supposons que l'identification de la structure externe a été réalisée. Considérons par exemple que les deux variables d'entrée X_1 et X_2 ont été choisies. À ce niveau, nous voulons déterminer les sous-ensembles flous associés aux deux variables d'entrée ainsi qu'à la variable de sortie Y ; sous-ensembles flous dont les combinaisons, également à déterminer, constituent la base de règles floues. Nous devons

finalement procéder à une identification des paramètres de ces ensembles flous en utilisant une procédure d'optimisation. L'ensemble de ces opérations constitue l'identification de la structure interne du système flou.

(3) *Approximation linguistique* : l'approximation linguistique se base sur l'utilisation de variables linguistiques (Zadeh, 1975). Elle consiste à associer à tout sous-ensemble flou une étiquette linguistique. L'étiquette en question est choisie de façon appropriée dans un dictionnaire prédéfini. Il est possible de lui rattacher un modificateur linguistique. Ceci est réalisé en utilisant des mesures de similarité entre les sous-ensembles flous et les étiquettes linguistiques. Dans ce cadre, différentes façons de faire ont été proposées (Sugeno and Yasukawa, 1993). L'approximation linguistique constitue en soi un niveau supérieur de compression de données. Elle permet de modéliser le système en utilisant un certain nombre de termes du vocabulaire humain, le but ultime étant d'arriver à une communication verbale homme-machine efficace.

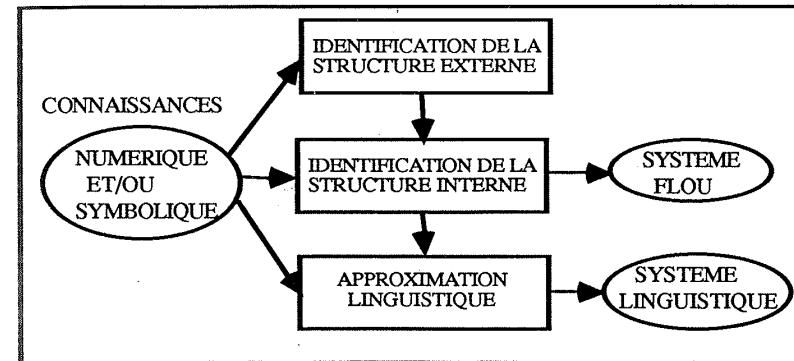


Figure 1. Méthodologie de modélisation d'un système flou.

Dans la suite de cet article, nous nous intéressons tout particulièrement à la seconde phase de la modélisation : l'identification de la structure interne.

III. IDENTIFICATION DE LA STRUCTURE INTERNE

L'identification de la structure interne du système flou revient à déterminer une collection de règles floues correspondant aux variables d'entrée et de sortie définies. Il faut donc déterminer les sous-ensembles flous formant

l'ensemble de ses règles et le nombre de règles. Deux grandes familles d'approches existent principalement dans la littérature :

- La première utilisant des techniques issues des réseaux de neurones artificiels et que l'on regroupe sous la terminologie de *modèles neuroflous*. Différentes manières d'utiliser des réseaux de neurones nous orientent vers différentes catégories de modèles neuroflous.

- La seconde utilisant des méthodes de classification. Plus particulièrement, elles sont couplés à une projection de l'espace de sortie sur l'espace des entrées pour l'induction de la base de règles floues.

Ces deux grandes familles de modèles, constituent la quasi-totalité des approches utilisées pour l'identification automatique de la structure interne des systèmes flous. La première s'oriente principalement vers une optimisation des paramètres, la seconde sur une détermination directe des règles floues. Néanmoins, la frontière entre les deux familles est loin d'être stricte et de nombreuses approches utilisent des idées communes aux deux familles.

La suite de cet article est dédiée à l'analyse d'une catégorie de modèles neuroflous. Une synthèse des différents autres modèles peut être trouvée dans Menage and Hartani, 1993.

IV. MODÉLISATION NEUROFLOUE : FORMALISME DANS UN CONTEXTE D'APPRENTISSAGE SUPERVISÉ

Dans cette section, nous analysons le problème de l'identification de la structure interne (sous-ensembles flous, partitions, structure et nombre de règles floues) d'un système flou à partir de techniques neuronales. Ceci regroupe ce qui est communément appelé *modélisation neurofloue* dans la littérature spécialisée. Nous nous intéressons en particulier à ce que nous avons présenté comme les systèmes flous à architecture neuronale. Dans la suite, ces modèles sont dénotés **SIFANE** (Système d'Inférence Floue à Architecture Neuronale).

Dans un premier temps, nous présentons une paramétrisation de systèmes d'inférence floue. Elle nous est utile pour développer une formalisation des modèles neuroflous de façon à regrouper de manière unifiée l'ensemble des démarches existantes. Cette formalisation est effectuée dans un contexte d'apprentissage supervisé. Elle nous permet de discuter la terminologie « neuroflou », de soulever certains problèmes pratiques et d'en proposer des solutions.

IV.1. Paramétrisation d'un SIF

Un SIF (Système d'Inférence Floue) est observé comme une fonction de transfert d'une partie E d'un espace de dimension n vers une partie F d'un espace de dimension m , il est défini comme suit :

$$\text{SIF} : E \subset R^n \rightarrow F \subset R^m$$

$$X = (x_1, \dots, x_n) \rightarrow Y = (y_1, \dots, y_m)$$

avec $Y = \text{SIF}_P(X)$ et $P = \{\text{paramètres du SIF}\}$

Dans la suite, dans un souci de simplicité et sans perte de généralité, on considère un système multientrée, monosortie dit MISO pour lequel $m = 1$.

Un SIF se base sur l'interprétation de symboles et de relations entre symboles. C'est pourquoi, nous caractérisons toute variable V (exemple : taille), définie sur un univers de discours U_V par un ensemble fini de symboles $S(V) = \{S_1, \dots, S_k\}$ (par exemple : grande, moyenne,...), les différents symboles étant associés à des sous-ensembles flous de fonctions d'appartenance $M(V) = \{M_1, \dots, M_k\}$ (par exemple : gaussiennes, triangulaires,...) définies sur U_V .

En reprenant la même description et en considérant un SIF de type MISO, les entrées et sorties du SIF se présentent comme suit :

$$X = (x_i; i = 1, \dots, n); \text{ à tout } x_i \text{ correspond } (U_{x_i}, \{S_{x_i, s}, s = 1, \dots, k_i\}, \{M_{x_i, s}, s = 1, \dots, k_i\})$$

$$Y = (y) \text{ auquel correspond } (U_y, \{S_{y, q}, q = 1, \dots, k_y\}, \{M_{y, q}, q = 1, \dots, k_y\})$$

avec k_i : nombre de sous-ensembles flous caractérisant la variable x_i

et k_y : nombre de sous-ensembles flous caractérisant y .

La fonction de transfert est interprétée selon une base de règles floues par une succession de trois mécanismes : une interface de fuzzification des entrées, un processus d'inférence floue et une interface de défuzzification des sorties.

a) Interface de fuzzification

L'interface numérique-symbolique de fuzzification est obtenue par l'intermédiaire d'une application associant à une observation numérique (généralement précise dans le cas des systèmes de commande), une partie

floue du domaine défini par les différents symboles. Cette application est définie par les différentes fonctions d'appartenance se présentant comme suit :

$$M_{x_i, s} : U_i \rightarrow [0, 1]$$

$$x_i(t) \rightarrow M_{x_i, s}(x_i(t))$$

avec : $x_i(t)$ observation à l'instant t ,

et chaque fonction d'appartenance $M_{x_i, s}$ étant définie par un ensemble de n_i ($n_i \in N$, $N = \{\text{entiers naturels}\}$) paramètres $a_{(i, s, m)}$, $m = 1, \dots, n_i$; en général $n_i = 2, 3$ ou 4 .

b) Processus d'inférence

L'obtention d'une loi de commande passe par la détermination de l'image de la composition des parties floues obtenues par l'intermédiaire de la procédure de fuzzification. Le mécanisme d'inférence passe par trois phases dont les deux dernières peuvent être dissociées ou associées selon le type de raisonnement flou choisi.

1. Calcul des valeurs de vérité

Soit une règle r dont l'antécédent est la conjonction de propositions « x_i est $S_{x_i, s(r)}$ », pour $i = 1, \dots, n$, $s(r)$ étant l'indice de la fonction d'appartenance relative à la règle r . On lui associe une valeur de vérité obtenue par agrégation de ses prémisses par un opérateur ϕ d'agrégation (une t -norme en général). Notons w_r la valeur de vérité de la règle r :

$$w_r = \phi_{1 \leq i \leq n} [M_{x_i, s(r)}(x_i(t))].$$

2. Relation prémisses-conséquent

Suite à l'activation de chacune des règles, un opérateur spécifique d'implication φ produit une transformation de la fonction d'appartenance correspondant à la partie conséquence des règles, de la forme :

$$M'_{y, q}(r) = \varphi[w_r, M_{y, q}]$$

et chaque fonction d'appartenance $M_{y, q}$ est définie par un ensemble de n_y ($n_y \in N$, $N = \{\text{entiers naturels}\}$) paramètres $b_{(q, p)}$, $p = 1, \dots, n_y$; en général $n_y = 2, 3$ ou 4 .

$M'_{y, q}(r)$ est l'image de la fonction d'appartenance $M_{y, q}$ obtenue par activation de la règle r .

3. Agrégation collective des règles

L'agrégation de l'ensemble des règles composant le SIF passe par une composition des différentes fonctions de l'étape précédente à travers un opérateur d'agrégation θ spécifique au type de raisonnement flou adopté (Bouchon-Meunier, 1991) (une t -norme ou une t -conorme selon le choix de φ). L'image de cette composition est :

$$M'_y = \theta_{r \in \text{ensemble de règles}} [M'_{y, q}(r)].$$

c) Interface de défuzzification

L'interface symbolique-numérique de défuzzification est observée comme une application qui associe à la partie floue obtenue par la procédure d'inférence une valeur numérique correspondant à la commande en sortie. L'approche de défuzzification D généralement utilisée est celle du centre de gravité qui se présente comme suit dans le cas d'une sommation discrète avec un pas de discrétisation k fini et $y(k)$ la valeur de y au pas k :

$$y = \frac{\sum_k M'_{y(k)} \cdot y(k)}{\sum_k M'_{y(k)}}.$$

d) Fonction paramétrable du SIF

La formalisation des différentes étapes d'un processus de commande floue amène à observer le SIF comme une fonction de deux familles de paramètres $P = \{P1, P2\}$, $P1$ caractérisée par le choix du raisonnement flou, soit des différents opérateurs spécifiques et $P2$ caractérisé par la définition des fonctions d'appartenance. Nous obtenons :

$$P1 = \{\phi, \varphi, \theta, D\} \quad \text{et} \quad P2 = \{A, B\}$$

$$\text{avec } A = \{a_{(i, s, m)}, m = 1, \dots, n_i; s = 1, \dots, k_i; i = 1, \dots, n\}$$

$$\text{et } B = \{b_{(q, p)}, p = 1, \dots, n_y; q = 1, \dots, k_y\}.$$

Le nombre de règles est induit par la combinaison des différents sous-ensembles flous des variables en prémisses des règles. La partition des univers de discours en sous-ensembles flous pour les différentes variables est fixée dans un premier temps. La forme géométrique des sous-ensembles flous reste quant à elle paramétrable.

IV.2. Identification par techniques neuronales

Différents résultats récents permettent d'établir la propriété d'approximateur universel pour de nombreuses classes de SIF, (Nguyen and Kreinovich, 1993), (Wang and Mendel, 1992). L'approximation revient à approcher avec une précision aussi bonne que possible, une fonction numérique continue, soit :

Pour toute fonction continue $f : X \rightarrow Y$ définie sur un compact $E \subset R^n$, à valeurs dans une partie $F \subset R^m$, on a :

$$\forall \varepsilon > 0, \quad \exists \text{SIF}_P / |\text{SIF}_P(X) - f(X)| < \varepsilon,$$

la notation $|\cdot|$ indiquant la distance euclidienne. Cette propriété garantit l'existence d'un SIF approchant la fonction f . Nous nous intéressons dans ce qui suit à la démarche qui permet l'identification du SIF en question. Le problème se formule comme suit :

Pour toute fonction continue $f : X \rightarrow Y$ définie sur un compact $E \subset R^n$, à valeurs dans une partie $F \subset R^m$, quel est le SIF de paramètres $P1$ et $P2$, tel que :

$$|\text{SIF}_{(P1, P2)}(X) - f(X)| \text{ est minimale}$$

Ceci revient à dire, en dénotant $P1^*$ et $P2^*$ les paramètres optimaux de $P1$ et $P2$: quels sont les paramètres $P1^*$ et $P2^*$ tels que :

$$\forall (P1, P2), |\text{SIF}_{(P1^*, P2^*)}(X) - f(X)| < |\text{SIF}_{(P1, P2)}(X) - f(X)|$$

Pour ce qui est de l'identification des éléments de $P1^*$, des études dressent des tableaux comparatifs en termes de stabilité et de robustesse mais aucune comparaison ne montre l'intérêt particulier d'une famille de paramètres par rapport à une autre. C'est tout particulièrement l'ensemble $P2^*$ que l'on veut identifier lors d'une modélisation neurofloue. Dans ce qui suit, nous formalisons la démarche utilisant les capacités d'apprentissage des réseaux de neurones pour l'identification de $P2^*$.

IV.3. Présentation du SIFANE

Nous considérons un réseau de neurones à couches et connexions directes. Dans ce cas la fonction de transfert est définie par :

$$\text{RN} : E \subset R^n \rightarrow F \subset R^m$$

$$X = (x_1, \dots, x_n) \rightarrow Y = (y_1, \dots, y_m)$$

avec $Y = \text{RN}_Q(X)$, $Q = \{\text{paramètres du RN}\}$

De façon explicite, nous avons : $Q = \{k, n_k, f_a, f_t, W\}$

avec k le nombre de couches,

n_k le nombre de neurones sur la couche k ,

W la matrice des poids synaptiques,

f_a la fonction d'activation du neurone

et f_t la fonction de transfert du neurone.

La démarche d'identification automatique de l'ensemble $P2$ est réalisée en donnant une architecture neuronale au SIF de façon à implémenter les différentes étapes du processus d'inférence floue et d'interpréter les éléments de $P2$ comme les poids à ajuster du réseau de neurones. Les fonctions d'activation f_a et les fonctions de transferts f_t des neurones correspondent aux fonctions d'appartenance et aux opérateurs logiques utilisés.

Comme précédemment souligné, différentes architectures neurofloues sont proposées dans la littérature (Horikawa *et al.*, 1990), (Berenji, 1992), (Wang and Mendel, 1992), chacune d'elles représentant un cas particulier de la description préalablement présentée. Le choix des fonctions de transfert et d'activation ainsi que du nombre de couches dépend bien évidemment du système flou choisi.

Afin d'en donner une illustration plus claire, nous proposons de modéliser un système flou complètement déterminé par les choix que nous effectuons. Nous prenons des fonctions d'appartenance gaussiennes identifiées par deux paramètres (moyenne, écart type) pour les sous-ensembles flous caractérisant les prémisses et pour les conséquences des règles (de manière réaliste, on néglige l'effet des deux parties symétriques de la gaussienne lors de la défuzzification, ce qui est équivalent à considérer des constantes en sortie, et donc un raisonnement flou du type Takagi-Sugeno), un approximateur différentiable de la t -norme « min » pour l'agrégation (Berenji, 1992) et le « min » comme pseudo-opérateur d'implication. La défuzzification est réalisée par la méthode du centre de gravité. Nous présentons la procédure d'identification des paramètres de l'ensemble $P2$ dans un contexte d'apprentissage supervisé.

Cette modélisation permet d'associer au SIF_(P1, P2) un RN à $k=4$ couches avec une matrice de poids W , avec $W = (W1, W2, W3, W4)$ et qui se présente comme suit :

couche 1 : $n_1 = n$ (dimension du vecteur d'entrée) et $W1 =$ poids fixes de valeur 1 ;

Cette couche permet la transmission des entrées au réseau.

couche 2 : $n_2 = \sum_{i=1}^n k_i$ et $W2 = (0, A)$ avec $A = \{a_{(i,s,m)}\}$ et

$$\text{card}(A) = 2 \sum_{i=1}^n k_i ;$$

Cette couche effectue le calcul des différents degrés d'appartenance. En prenant des fonctions d'appartenance gaussiennes, toute fonction d'appartenance aura pour paramètres sa moyenne et son écart type.

couche 3 : $n_3 =$ nombre de règles $\leq \prod_{i=1}^n k_i$ et $W3 =$ poids fixes ($= 1$ ou 0) ;

Cette couche permet le calcul de la valeur de vérité de chacune des règles.

couche 4 : $n_4 = 1$ (dimension du vecteur de sortie) et $W4 = (B)$ avec $B = \{b_{(q,p)}\}$ et $\text{card}(B) \leq \prod_{i=1}^n k_i$.

Cette dernière couche réalise l'agrégation des règles et la procédure de défuzzification.

La figure 2 présente l'architecture neuronale obtenue :

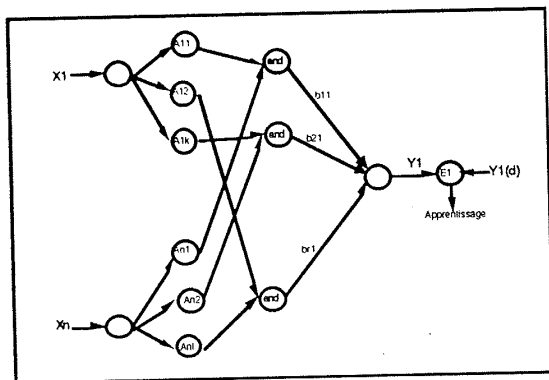


Figure 2. Architecture neuronale du système flou : SIFANE.

Cette association nous permet d'établir que la représentation générale du SIF peut prendre une architecture similaire à celle d'un réseau de neurones. L'architecture de ce dernier, noté SIFANE, demeure particulière : les différents poids synaptiques, les fonctions d'activation et des fonctions de transfert des différents neurones sont associés respectivement aux paramètres des fonctions d'appartenance et aux différents opérateurs logiques. Le nombre de neurones sur l'ensemble des couches est bien définie par le nombre de règles floues (couche 3) et le partitionnement des univers de discours des variables (couche 2).

IV.4. Pourquoi neuroflou ? Est-ce indispensable ?

Nous avons présenté l'architecture neuronale faisant une analogie avec la structure du SIF. Une question importante se pose, question sujette à de multiples débats. Pourquoi la terminologie « neuroflou » pour ce type de modèle et pourquoi rechercher une architecture neuronale équivalente au SIF ? À vrai dire, à l'origine nous avons un SIF et à la fin nous obtenons toujours un SIF. Cela dit, le fait de donner au SIF une analogie neuronale en terme de structure, permet de tirer profit de l'ensemble des techniques développées dans le domaine du neuronal. Les techniques d'apprentissage de type supervisé, non supervisé ou par renforcement peuvent être appliquées avec de simples modifications d'algorithmes pour l'identification des paramètres du SIF. Les techniques d'élagage de neurones et d'élagages de connexions peuvent être également appliquées pour la détermination du nombre de règles et du partitionnement des univers de discours. Aurait-on pensé utiliser les algorithmes d'apprentissage par renforcement depuis longtemps utilisés par la communauté neuronale pour l'identification de SIF s'il n'y avait pas cette interaction ? Peut-être pas et sûrement bien plus tard.

Cette appellation est-elle indispensable ? Nous pensons que non. Il est toujours possible de parler de SIF adaptatifs ou encore de procédures d'ajustement des paramètres du SIF. Mais dans ce cas, les réseaux de neurones ne sont-ils pas également des systèmes adaptatifs, utilisant des algorithmes d'adaptation appropriés ? De ce fait, même si l'appellation neurofloue n'est pas indispensable, elle est à notre avis très utile, ne serait-ce que pour faire profiter l'une des techniques des avantages de l'autre. Disons que la même question pourrait être posée sous la forme : les réseaux de neurones artificiels sont-ils vraiment une analogie des réseaux de neurones biologiques ? Beaucoup répondent par la négative. Néanmoins, la grande partie des algorithmes d'apprentissage trouvent leur origine dans

les mécanismes biologiques du cerveau. Le fait de faire évoluer les choses par analogie est donc d'une grande utilité.

Finalement, nous pensons que le fait d'évoluer par analogie est utile. Il l'est d'autant plus qu'il permet de faire progresser à la fois les deux techniques floue et neuronale. Quant à la terminologie, comme son nom l'indique, ce n'est qu'une convention d'écriture permettant d'harmoniser les choses. La terminologie neurofloue est bien reconnue dans la littérature spécialisée depuis plus de cinq ans.

V. APPRENTISSAGE SUPERVISÉ DU SIFANE

V.1. Problématique de l'apprentissage supervisé de SIFANE

Nous pouvons établir une équivalence entre le SIF_{P1,(A,B)} et le RN_W. Il est donc possible d'écrire :

$$\text{SIF}_{P1,(A,B)} = \text{RN}_W,$$

l'égalité traduisant la notion d'équivalence entre systèmes. L'utilisation des notions d'apprentissage pour la détermination de W optimal, soit W^* , est donc équivalente à la détermination de A et B optimaux, soit A^* et B^* . Une approche classique basée sur la rétropropagation du gradient nous permet d'effectuer l'apprentissage en partant d'une base de données (X, Y^*) . En tenant compte de $W3$ et de $W1$, nous aurons à dimensionner $W2$ et $W4$.

Nous nous plaçons dans le cadre de l'apprentissage supervisé. Cette problématique est bien étudiée dans le domaine du neuronal (Bochereau, 1991). Par simple analogie, nous analysons la problématique pour ce qui est de l'identification des paramètres optimaux du SIF. Comme précédemment souligné et étant donné l'architecture neuronale donnée au SIF, la démarche d'identification de $P2$ et donc de W revient à résoudre :

$$Y = \text{RN}_W(X) + \varepsilon$$

X désignant donc le vecteur des entrées, E_X l'espace vectoriel associé, Y le vecteur de sortie, E_Y l'espace vectoriel associé et ε la précision de l'approximation.

Le produit des espaces vectoriels des vecteurs X et Y est muni d'une distribution de probabilité $p(X, Y)$. Cette distribution permet de définir un critère de coût, fonction d'un certain nombre d'entrées-sorties définissant la

base d'apprentissage du modèle. Ce critère de coût C_{RN} associé au réseau de neurones RN a la forme générale suivante :

$$C_{\text{RN}} = \int_{E_X} \int_{E_Y} \pi(\text{RN}_W(X), Y) p(X, Y) dX dY$$

qui est donc égal au critère de coût associé au SIF :

$$C_{\text{SIF}} = \int_{E_X} \int_{E_Y} \pi(\text{SIF}_{(P1, P2)}(X), Y) p(X, Y) dX dY$$

avec $\pi(., .)$, une fonction à valeurs dans R^+ . En général, $\pi(., .)$ est une fonction de coût quadratique.

L'objectif de l'apprentissage supervisé est donc de déterminer W^* , ou encore $P2^*$ tels que C_{RN} et C_{SIF} atteignent leur minimum local. Nous parlons donc de convergence vers ce minimum local. Le critère coût est impossible à calculer de façon directe, la distribution de probabilité étant *a priori* inconnue. Il est donc estimé à partir de l'ensemble d'apprentissage. Il est utile de préciser que ce processus d'apprentissage provoque des erreurs qui biaisent le choix du SIF optimal. Trois catégories principales d'erreurs existent (Bochereau, 1991) :

- erreur d'échantillonnage : elle est due à l'approximation de l'intégrale par une somme finie sur un ensemble fini d'observations ;
- erreur due au choix de la structure de l'approximateur, c'est-à-dire au nombre de paramètres à ajuster. Ceci est directement lié à l'architecture du réseau, qui dans notre cas, est fonction du nombre de règles floues et de la partition des différents univers de discours. En fonction de ce nombre de paramètres et du nombre d'exemples, le problème d'identification est sous-déterminé, exactement déterminé ou sur-déterminé ;
- erreur due à la procédure d'optimisation. Les algorithmes procèdent généralement par tâtonnement et de manière itérative jusqu'à l'atteinte de la solution localement optimale. Différents facteurs peuvent causer une non convergence de l'algorithme.

Dans la pratique et dans la version de base de l'algorithme, la matrice des poids synaptiques W se détermine par une rétropropagation du gradient de l'erreur à la sortie du réseau.

L'identification de W étant équivalente à l'identification de $P2$, le SIF optimal au sens du critère coût choisi est donc déterminé.

V.2. Les problèmes posés par l'apprentissage du SIFANE

Dans une optique d'apprentissage supervisé, des algorithmes dérivés de la rétropropagation du gradient conviennent. Certains problèmes, autres que ceux directement liés aux modalités d'apprentissage soulevés auparavant, demeurent cependant posés :

(1) – les problèmes classiques aux algorithmes neuronaux : ils sont liés aux choix de bases d'apprentissage et de généralisation et taux d'apprentissage à ajuster.

(2) – les problèmes liés à l'interprétabilité des résultats obtenus : en effet, il est important de pouvoir interpréter de manière explicite les règles obtenues. Dans le cas contraire, nous obtiendrons un simple approximateur de fonctions sous forme de boîte noire, pour lequel aucune interprétation explicite des connexions et poids du réseau n'est possible. Le principe bien connu de l'utilisation de méthodes d'optimisation pour l'ajustement des poids repose sur l'utilisation de fonctions de coût globales. Autrement dit, il est possible d'avoir de bonnes performances en termes d'erreur globale d'apprentissage/généralisation avec un mauvais comportement local, c'est-à-dire que les règles floues obtenues ne traduisent pas nécessairement le comportement local désiré du système. Les problèmes que nous avons constaté expérimentalement, sont également bien soulevés et illustrés dans Babuska *et al.*, 1994. Ceci nous amène à nous intéresser aux procédures d'initialisation des paramètres. En effet, il est nécessaire de bien initialiser les paramètres des fonctions d'appartenance, c'est-à-dire à des valeurs correspondant à un comportement local convenable. L'optimisation ultérieure aura pour but pour but d'ajuster ces paramètres sans grande modification du comportement local.

(3) – les problèmes liés au choix du nombre de neurones sur chacune des couches, c'est-à-dire de la partition de l'univers de discours des différentes variables et des règles floues composant le SIF. En fait, ces choix sont dépendants des procédures d'optimisation des paramètres et il est difficile de séparer les différentes opérations. À ce niveau, peu de travaux abordent le problème dans le cadre des modèles neuroflous. Nous soulignons les travaux récents (Lin et Lee, 1991) utilisant un apprentissage non supervisé pour la création de neurones correspondants aux différentes règles ainsi que ceux de Berenji, 1993 utilisant un regroupement dans l'espace produit entrée-sortie. Comme dans le cas des réseaux de neurones, aucune méthode exacte ne permet de donner des solutions directes. En général, on procède par tâtonnement jusqu'à la détermination de la meilleure solution. Dans le contexte de modèles neuroflous, nous proposons de tirer profit de la nature

explicite de l'information distribuée au travers du réseau, couplée à des procédures d'élagage de neurones et de connexions, pour l'approche de ce problème. Elle nous amène à présenter des démarches heuristiques ayant pour objectif une identification itérative de la structure interne du système flou.

V.3. Apprentissage de SIFANE : Méthodologie présentée

L'apprentissage des réseaux neuroflous que l'on trouve dans la littérature, est généralement réalisée en utilisant des adaptations de l'algorithme de rétropropagation. Dans cette section, nous voulons tenter d'apporter des éléments de réponse aux problèmes posés. Tout particulièrement celui de l'interprétabilité des règles.

Nous introduisons des variantes aux algorithmes d'apprentissage existants, dont nous nous inspirons, Alche Buc, 1993, Lin and Lee, 1991, Glorennec, 1993, Berenji, 1992. Nous présentons une méthodologie pour l'apprentissage des SIFANE, tout en laissant un certain nombre d'opérations au choix du concepteur (forme géométrique des sous-ensembles flous, partitions,...). Les points essentiels concernent donc : l'initialisation des paramètres et la réduction de la dimension de la base de règles. Une architecture spécifique est choisie pour les expérimentations. Nous présentons nos motivations pour les choix effectués ainsi qu'un ensemble d'exemples de validation.

a) Initialisation des paramètres et de la structure interne du SIF

Il est important de procéder à une bonne initialisation de la structure interne du système flou pour deux raisons principales :

– la première est directement reliée aux performances de l'apprentissage. En effet, il a été précédemment souligné que l'une des principales motivations à utiliser des modèles neuroflous réside en la recherche d'un bon apprentissage/généralisation du système flou, c'est-à-dire, d'une fonction erreur minimale. Il est clair qu'une bonne initialisation donnera une solution initiale plus proche de l'optimum local et donc la convergence est à la fois plus rapide et mieux garantie ;

– la seconde est reliée au problème de l'interprétabilité de la base de règles floues obtenue, telle que nous l'avons auparavant discuté. En effet, une bonne initialisation permet d'initialiser les différents paramètres des fonctions d'appartenance dans un voisinage des paramètres optimaux. Leur interprétabilité est de ce fait mieux maîtrisée.

L'importance de l'initialisation soulignée, il nous reste à définir les procédures. Deux manières sont à distinguer, selon la nature des informations disponibles. L'une manuelle, l'autre automatique, introduisant à la fois des procédures d'apprentissage non supervisé et une démarche heuristique. Nous les présentons de façon générale (sans considération de fonctions d'appartenance particulière et d'opérateurs logiques spécifiques) en focalisant sur la seconde qui nous intéresse particulièrement :

– la première est donc *manuelle*. Elle consiste à extraire directement la connaissance des experts sous forme symbolique. Elle revient tout simplement à introduire directement les différentes règles (nombre de sous-ensembles flous et les différentes combinaisons) ainsi que les paramètres des fonctions d'appartenance, sous forme d'un codage de la matrice des poids du SIFANE,

– la seconde, *automatique*, nous intéresse plus particulièrement dans la suite. Le but est de réaliser une assez bonne initialisation des paramètres en utilisant l'information disponible. Différentes phases sont à distinguer. L'initialisation des prémisses par regroupement flou, des conséquences par heuristique ainsi que la façon de combiner les sous-ensembles flous constituant les différentes règles du système flou. Nous détaillons les différentes étapes :

Initialisation des prémisses des règles

Nous proposons d'utiliser une auto-organisation des données en utilisant l'algorithme des Fuzzy C-Means (FCM) (Bezdeck, 1981) afin d'obtenir un partitionnement flou pour chacune des variables d'entrée. L'idée d'utiliser un apprentissage non supervisé a préalablement été développée dans l'étude des réseaux de neurones, pour l'initialisation de réseaux RBF (Moody et Darken, 1988). Elle a également été reprise pour ce qui est des réseaux neuroflous avec l'utilisation de l'algorithme de Kohonen dans Lin and Lee, 1991.

Nous ne réalisons pas de regroupement flou sur l'ensemble de l'espace des entrées de manière à ne pas avoir à réaliser une projection sur chacune des coordonnées, ce qui permet d'éviter des sources d'erreurs. Le partitionnement proposé permet de représenter de la meilleure manière possible les exemples de la base d'apprentissage en termes de sous-ensembles flous. Le nombre de classes peut quant à lui, soit être fixé, soit être déterminé par une validation du clustering (Bezdeck, 1981).

Il est raisonnable de penser qu'un partitionnement flou utilisant les FCM est préférable du moment qu'il nous permet d'obtenir l'ensemble des paramètres la caractérisant, grâce à une approximation par la représentation géométrique choisie (gaussiennes, triangulaire, etc.). Le clustering flou permet également de produire plus de sous-ensembles flous dans les régions à forte densité de

données et donc plus de règles floues dans ces mêmes régions. Ceci n'est évidemment pas possible avec un autre mode de partitionnement.

Remarque : Le résultat du regroupement est un sous-ensemble flou défini de manière discrète. Il est donc nécessaire de l'approcher par la représentation géométrique que l'on aura choisie.

Initialisation des conséquents des règles

Une procédure heuristique est considérée. Elle se base sur l'idée suivante : considérons une règle donnée et déterminons un vecteur qui l'active totalement (degré de vérité égal à 1). De ce fait, le sous-ensemble flou correspondant à la variable de sortie pour cette règle aura un poids important lors de la défuzzification. Les autres sous-ensembles flous auront un poids plus faible. L'image de ce vecteur doit intuitivement être proche de la sortie voulue du système flou. Cela signifie que l'on peut associer de manière approximative à ce vecteur, une sortie égale au centre de gravité de ce sous-ensemble flou. Partant de cette idée intuitive, on détermine l'exemple de la base d'apprentissage qui se rapproche le plus du vecteur activant totalement la règle. L'initialisation du sous-ensemble flou de sortie est faite en lui donnant comme centre de gravité, l'image du vecteur trouvé. Ce processus est répété pour l'ensemble des règles.

Une initialisation dans ce même ordre d'idée a été faite en prenant pour chaque règle i , $b_i = y_k^*$, avec k indice de l'exemple tel que la valeur de vérité w_i de la règle i vérifie $w_i(k) = \max(w_i(l))$, $l = 1, \dots, Nb$, Nb étant le nombre d'exemples (Glorennec, 1993). Dans notre cas, nous proposons de procéder comme suit :

- pour toute règle i , déterminer le vecteur X de R^n (n étant la dimension de l'espace d'entrée) tel que $w_i(X) = 1$. Il suffit de prendre un élément du noyau des sous-ensembles flous composant la prémisse de la règle i ;
- identifier l'exemple k de la base d'exemples, le plus proche (au sens d'une distance euclidienne) du vecteur X et poser $b_i = y_k^*$, b_i est considéré comme le centre de gravité du sous-ensemble flou de sortie ;
- reconstruire un sous-ensemble flou possédant la représentation géométrique choisie à partir de son centre de gravité b_i . Par exemple, considérer le centre de gravité comme un centre de symétrie.

L'approche que nous proposons a l'avantage d'être plus générale. Elle est indépendante de la l -norme choisie pour l'agrégation des prémisses et de la façon de combiner les règles. Elle permet d'initialiser les poids du SIFANE sans présentation des exemples. Aussi, elle donne plus d'informations sur les sous-ensembles flous de sortie.

b) Procédures d'apprentissage

L'apprentissage consiste à utiliser des variantes de l'algorithme de rétropropagation du gradient, adaptées aux fonctions utilisées. Le choix des taux d'apprentissage et des termes de moment pour chacun des paramètres à ajuster est à effectuer. Les travaux auparavant réalisés dans le cadre de l'apprentissage de réseaux RBF, montrent l'intérêt de réaliser l'apprentissage en plusieurs phases (Musavi, 1992). Une fois l'initialisation réalisée, l'algorithme à utiliser se présente comme suit :

- rétropropagation du gradient sur la dernière couche seulement pendant un nombre d'itérations fixé ;
- rétropropagation du gradient sur l'ensemble des couches avec arrêt quand le terme d'erreur relatif à la base de test ou la valeur du gradient devient inférieur à un seuil défini.

Le schéma suivant présente une synthèse de la méthodologie d'apprentissage :

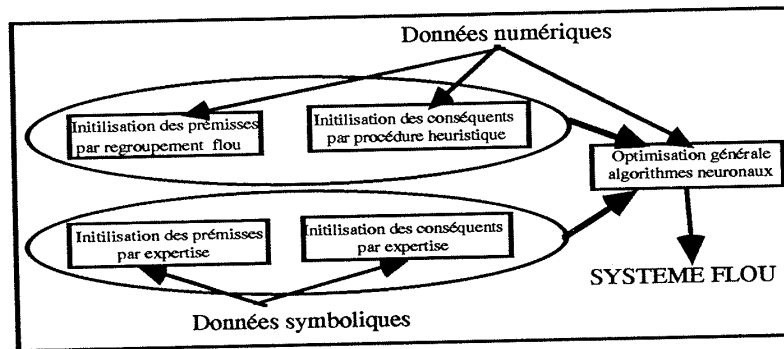


Figure 3. Schéma synthétique de la méthodologie d'apprentissage.

La méthodologie présentée pour la conception de systèmes neuroflous demeure générale. C'est-à-dire indépendante de la structure du système flou choisi. Néanmoins, les algorithmes d'apprentissage utilisés doivent être adaptés en fonction de différents paramètres (forme des sous-ensembles flous, opérateurs logiques...). Dans notre cas, l'architecture du SIFANE choisi nous a permis d'illustrer, suite à un certain nombre d'expérimentations (Hartani et al., 1994), l'intérêt de la méthodologie présentée.

Remarque : Entre les deux approches présentées, manuelle ou automatique, peuvent exister un certain nombre de variantes. Par exemple,

l'initialisation des prémisses par expertise et des conséquents automatiquement. Il s'agit de choisir la meilleure façon de faire en fonction de l'information disponible.

VI. CONCLUSION

La motivation principale de cet article a consisté à proposer des investigations dans le domaine de la modélisation neurofloue. Nous avons présenté une méthodologie générale pour la modélisation de systèmes flous par modèles neuroflous. Une formalisation des approches neurofloues, dans un contexte d'apprentissage supervisé, nous a permis d'unifier les différentes architectures neurofloues similaires. Nous avons établi la paramétrisation d'un SIF et montré qu'il pouvait être approché par un réseau de neurones avec une architecture multicouches particulière. Nous avons, par la suite, montré comment les capacités d'apprentissage des réseaux de neurones permettent l'identification des paramètres optimaux du système SIF.

Bibliographie

- F. ALCHE BUC, Modèles neuronaux et algorithmes constructifs pour l'apprentissage de règles de décision, Thèse de l'Université Paris-Sud, décembre 1993.
- R. BABUSKA, R. JAGER et H. B. VERBRUGGEN, *Interpolation issues in Sugeno-Takagi reasoning*, IEEE conference on fuzzy systems, Orlando, USA, 1994.
- H. R. BERENJI et P. KHEDKAR, Learning and tuning fuzzy logic controllers through reinforcements, *IEEE trans on Neural Networks*, Vol. 3, n° 5, 1992.
- J. C. BEZDEK, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.
- L. BOCHEREAU, Conception et interprétation de réseaux connexionnistes multicouches, thèse de doctorat de l'Université Paris-VI, décembre 1991.
- B. BOUCHON-MEUNIER, *Various logical approaches in fuzzy control*, Actes journée nationale commande floue, CNRS, Paris, 1991.
- P. Y. GLORENNEC, Algorithmes d'apprentissage pour réseaux neuroflous, Rapport interne IRISA, avril 1993.
- R. HARTANI, S. HAYAT et P. GALLINARI, Modélisation neuronale d'un système d'inférence floue : expérimentations et extensions, *Proc. of International Conf. on Information Processing and Management of Uncertainty*, Paris, France, 1994.
- S. I. HORIKAWA, T. FURUHASHI, S. OKUMA et Y. UCHIKAWA, A fuzzy controller using a neural network, *Proc. of International Conf. on Fuzzy logic, Neural networks and Soft Computing*, Iizuka, Japan, 1990.
- C. T. LIN et C. S. G. LEE, Neural network based fuzzy logic control and decision systems, *IEEE trans. Computers*, n° 40, December 1991, p. 1321-1335.

- X. MENAGE et R. HARTANI, Synthèse de l'association des techniques floues et neuronales, Rapport interne du LAFORIA, mai 1993.
- J. MOODY et C. DARKEN, Learning with localized receptive fields, *Proc. of the Connectionist Models Summer School*, Ed. Touretzky, Morgan Kaufman, 1988, p. 113-143.
- M. T. MUSAVI, W. AHMED, K. H. CHAN, K. B. FARID et D. M. HUMMELS, On the training of radial basis function classifiers, *Neural Networks*, Vol. 5, 1992, p. 595-603.
- H. T. NGUYEN et V. KREINOVICH, *On approximation of controls by fuzzy systems*, Proc. Fifth IFSA, 1993, p. 1414-1417.
- A. RALESCU et R. HARTANI, Some investigations in fuzzy and linguistic modeling, *Proc. of IEEE/IFES conf. on fuzzy systems*, Yokohama, Japan, 1995.
- M. SUGENO et T. YASUKAWA, A fuzzy logic based approach to qualitative modeling, *IEEE trans. on Fuzzy systems*, Vol. 1, 1993.
- L. X. WANG et J. MENDEL, Fuzzy basis functions, universal approximation and orthogonal least square learning, *IEEE trans. on Neural Networks*, Vol. 3, n° 5, September 1992.
- L. A. ZADEH, The concept of linguistic variable and its application to approximate reasoning, *Information Science*, Vol. 8, 1975.

UN MODÈLE DE SIMULATION DES COMPORTEMENTS DYNAMIQUES DES PROCESSUS DE MOTIVATION*

M. KARSKY ¹, G. DONNADIEU ²,
St. COPIN ¹, S. PITARCH ¹, J. FOURCADE ³

Résumé

Le modèle MODMIL constitue une synthèse de deux modèles intégrant des théories de la motivation et des processus psychologiques validés expérimentalement.

Il s'agit de concevoir et d'analyser la motivation sous l'angle systémique et de réaliser un simulateur permettant une mise en pratique et une vérification de l'évolution temporelle des concepts introduits et des comportements qui en résultent.

Un exemple d'application sur un cas concret de démotivation, permet d'accéder à une compréhension du comportement de l'individu, à la génération de solutions envisageables et à une aide au choix d'une solution adéquate.

En conclusion, les auteurs donnent des indications d'utilisation du simulateur.

Abstract

MODMIL is a model which is a synthesis between two models which integrate theories on motivation and experimentally verified psychological processes.

Motivation is viewed as a general system, the practical propose being to realize a simulation tool which would permit the analysis of the numerous concepts introduced in the model and the resulting behaviors in time.

A practical application is described, showing a demotivation behavior of a person, giving a better understanding of the process, suggesting and

1. K.B.S. (Knowledge Based Simulation), 287, rue Saint-Jacques, 75005 Paris.

2. Entreprise & Personnel, 48, route de la Reine, 92100 Boulogne.

3. DGA/DAT, Bureau Prospective Long Terme, BP 24, 92211 Saint-Cloud.

* Le travail décrit dans cet article a pu être effectué grâce à l'aide de ELF Aquitaine (MODÉRÉ) et de la DGA/Direction des Armements Terrestres (MODMIL).