Neuro-Fuzzy Based Approach For Hybrid Force/Position Robot Control

Y. Touati , K. Djouani and Y. Amirat
Computer Science and Automation Lab - LIIA, University of Paris Val de Marne
122, rue Paul Armangot – 94400 Vitry / Seine, FRANCE
Phone: +331-4180-7382 Fax: +331-4180-7369 E-mail: touati@univ-paris12.fr

Abstract

This paper presents neuro-fuzzy control approach for MIMO systems. Motivated by the hybrid force/position control of robot manipulator problem, a systematic design procedure for fuzzy rules generation and optimization is proposed. The proposed neuro-fuzzy controller is constructed with respect to three phases. In the first one, which is called parameters learning phase, the neuro-fuzzy system is considered as feed-forward neural network and back-propagation learning algorithm is then applied for parameters identification in order to map Input/Output data. In the second phase, a new clustering algorithm based on the inclusion concept is used for optimal clusters identification. Finally, the fuzzy rule base is generated and optimized. A 2 DOF planar manipulator force/position control simulation is presented and results discussed.

Keywords: Neuro-fuzzy control, Optimization and Clustering, Hybrid Force/Position Control.

1. Introduction

Identification and control of complex systems have been a difficult challenging problem to be solved for decades. Due to the lack of classical systems theory for representing complex systems characteristics, the exact dynamics of such systems are usually unknown so that it's difficult to satisfy the requirements of stability and desired performance of the tasks. So, suitable control algorithms are needed in handling various complex situations. Such tasks require accurate positioning, which depends upon the accurate estimations of robot kinematics and dynamics. Moreover, industrial tasks are also concerned within force control and mostly unknown robot/environment interaction problems. That constrained motion control requires a controller that combines both position and force control, which is called hybrid force/position control [1][9]. However, some knowledge about the process behaviors may exists in many different forms, e.g., simplified mathematical models which are valid only within limited operating ranges, heuristic rules which attempt to describe observed system behavior, etc. Such knowledge leads to some quantitative and qualitative information that can be used in the control design problem by taking into account system complexity and uncertainty. Integrating these various forms of information into a control system is and stills a challenging problem [3]. Neural networks and fuzzy logic provide learning and rule based reasoning that can be applied to the control of complex systems whose dynamic model description is either too complex or unavailable. They may add such abilities. In this respect, there have been recent many research works suggesting that various forms of combined uses of fuzzy logic and neural networks are complementary techniques [9][11][12]. The fundamental concept of such hybrid system is to overcome each other's weakness, leading to a new approach to solve problems. Many researchers are currently investigating ways and means of building neuro-fuzzy systems incorporating the notion of fuzziness into a neural networks framework.

In this paper, an hybrid Neuro-Fuzzy control approach is presented and the learning algorithms are developed. This approach is based on a neuro-fuzzy controller design methodology, which proceeds in three phases. In the first one, the fuzzy logic system is considered as a feed-forward neural network back-propagation on which a learning algorithm is applied in order to map some input/output data. This phase is called parameters learning phase. In second phase, a new clustering algorithm based on the inclusion concept is used for optimal clusters identification [9]. Finally, the resulting fuzzy rule base is generated and optimized according to conflict and redundancy analysis.

2. Neuro-fuzzy Approach for System modeling and Control

In this section, the design of a Neuro-Fuzzy Controller (NFC) is developed for a large-scale complex systems. This approach consists on the decoupling of the NFC system into a several Multi-inputs/Single-output NFC sub-systems (MISO) [12]. The number of the NFC sub-systems is equal to MIMO controller outputs dimension. The approach is based on Jang's ANFIS model for function approximation [8]. Fig.1 shows an external force control that uses a NFC. The network consists of several sub-systems called NFC modules, where each module is configured according to its own inputs/output data and optimization criteria.

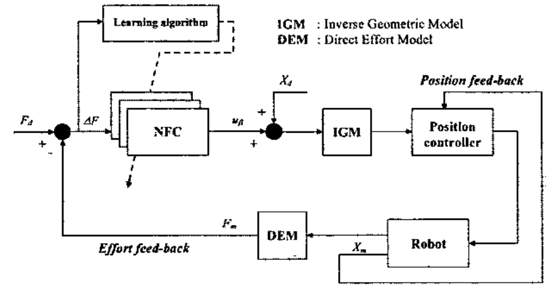

Fig.1 External Force Control using NFC

The learning algorithm based on back-propagation method is applied to adjust the NFC parameters. The variables F_d and F_m represent respectively the desired and measured efforts, X_d, X_m, the desired and measured positions, and $u_{fl} = (u_1, u_2, .., u_n)$ the controller output. IGM and DEM are respectively the Inverse Geometric Model and direct effort Model. The architecture illustrated on Fig.2 represents a single MISO NFC module, which is considered as a neuro-fuzzy system built from four-layer feed-forward network. Each fuzzy module takes the antecedents ΔF_k at its inputs and produces an action u_k, where k = 1,.., n, with n, number of NFC modules.

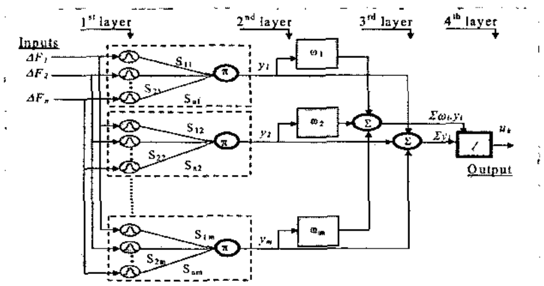

Fig.2 MISO Neuro-fuzzy Controller Architecture

As in ANFIS model, this architecture implements rules of the following form [13]:

$$R_j : If\ \Delta F_1\ is\ A_{1j}^{(1)}\ and \cdots and\ \Delta F_n\ is\ A_{nj}^{(j)}\ then\ \ y_k = f(\Delta F_1, \cdots, \Delta F_n)$$

where $A^{(i)}_{ij}$ represents the fuzzy sets.

The NFC design passes by the partitioning of each input/output variable space in several fuzzy sub-sets. Firstly, the Gaussian membership functions type (MFs) are chosen and a free partition is applied to the universe of discourse. From some observations of ΔF_i, the fuzzy inference consequence y_k obtained by simplified fuzzy reasoning method. The membership functions for each input variable ΔF_i, can be written as follows:

$$A_{ij}^{(k)} = S_{ij}^{(k)}(\Delta F_i) = exp\left(-\frac{1}{2}\left((\Delta F_i - m_{ij})/\sigma_{ij}\right)^2\right) \qquad (1)$$

The inputs/output relationships for the k^{th} NFC module are given by [12] :

$$u_k = \frac{\sum_{i=1}^{n}\left(w_j \cdot \prod_{j=1}^{k} S_{ij}^{(k)}(\Delta F_i)\right)}{\sum_{i=1}^{n}\left(\prod_{j=1}^{k} S_{ij}^{(k)}(\Delta F_i)\right)} \qquad (2)$$

where ΔF_i is the input variable, m_{ij} and σ_{ij} the i^{th} mean and standard deviation respectively of the j^{th} rule.

3. Adaptive NFC Design Methodology

The design of an adaptive NFC is considered as a fuzzy sets learning problem [10]. In our approach, it can be performed in three phases: parameters/structure learning, structure learning and real time parameters adaptation.

3.1 Parameters/Structure Learning

Assuming that prior knowledge is available in a certain form (input/output database), at this stage, we cater with NFC parameters identification. The goal of the learning algorithm is to modify the NFC membership functions and consequent parameters in order to optimize some global behavior criteria or to map some inputs/output data. In this paper, we are concerned with supervised learning, so the NFC system is viewed as a feed-forward neural network and back-propagation learning algorithm is applied off line to map some input/output data. As shown on Fig. 3, the NFC system maps the forward dynamics of the conventional controller. The aim of the learning algorithm is to update the parameters $Par=(m_{ij},\sigma_{ij},w_j)$, where (m_{ij},σ_{ij}) are the membership functions parameters and w_j, the consequent parameters. According to the back-propagation algorithm, the parameters updating equation is given by [12]:

$$Par(k+1) = Par(k) - \eta_{par}\frac{\partial E(k)}{\partial Par(k)} \qquad (3)$$

where η_{par} is the learning rate, k the learning iteration number, u_d and u_{fl} respectively the desired output value and the corresponding fuzzy result and E an objective function given by :

$$E = (\ u_{fl}(k) - u_d(k))^2 / 2 \qquad (4)$$

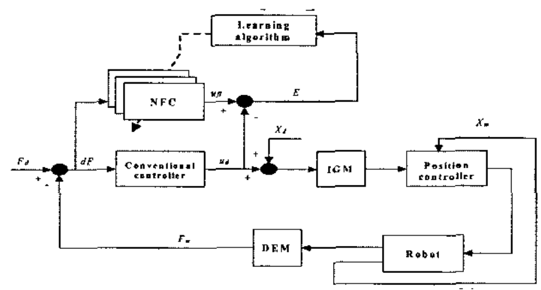

Fig.3 Identification Structure Scheme

3.2 Structure Learning

In this phase, we cope with the generated fuzzy rules obtained after the parameters learning phase. In general, this rule-base suffers from redundancy and conflicts of data, most of which are less useful [9]. This redundancy is often present in the form of similar MFs in the premise of the resulting rule-base. Such similarity within fuzzy sets render difficult to attach qualitatively meaningful linguistic labels to the different MFs. The high number of MFs makes difficult to obtain the meaning of the model, and thus the working of system at hand. A semantically unclear model is not easily verified after design phase for the model. Consequently, a simplification phase allowing the elimination of redundancy is required. For this purpose, we cope with fuzzy clustering [9][12] based on inclusion concept where the rule-base has to be simplified. This simplification occurs in the sense that similar MFs pertaining to the premise of fuzzy rule-base are merged and replaced by one common MF, capturing the meaning of the former. So, we propose an algorithm based on the class of fuzzy clustering method introduced by Bezdek [4] and an inclusion concept, which is characterized by an inclusion index I_d [2][5] [6][9].

3.2.1 Inclusion Index Construction

Let us denote by G the Gaussian distribution characterized by $[m,\sigma]$. In general, 97% of the information supplied by the distribution is concentrated into the interval $[m-3\sigma, m+3\sigma]$. $I_d(G_{ij},G_k)$ stands for the degree of inclusion of the Gaussian (m_{ij},σ_{ij}) in (m_k,σ_k), and characterized by the surface formed by the intersection of G_{ij} and G_k, $(G_{ij}\cap G_k)$ (Fig. 4). The Gaussian G_{ij} and G_k can be represented respectively as a circle centered in m_{ij} and m_k, and with a radius σ_{ij} and σ_k. Clearly, asserting that Id should depend on the following parameters: m_{ij}-m_k and $3.(\sigma_{ij}+\sigma_k)$.

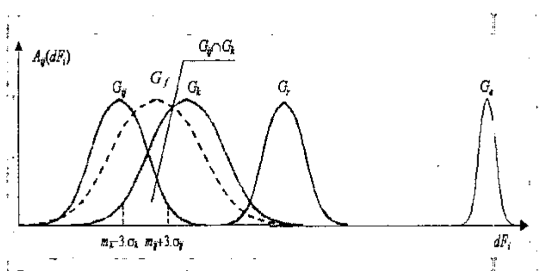

Fig.4 Fuzzy Set Overlapping

In the same manner, as it can be noticed from Figure 4, $I_d(G_{ij},G_k) \rightarrow 0$ if $(m_k - 3.\sigma_k) > (m_{ij} + 3.\sigma_{ij})$, which mean that the two distributions are far away from each other and $G_{ij}\cap G_k \rightarrow \varnothing$. The positive value of $I_d(G_{ij},G_k)$ is reached when $(m_{ij} + 3.\sigma_{ij}) > (m_k - 3.\sigma_k)$ and $G_{ij}\cap G_k \neq \varnothing$. This can better be interpreted as :

$$I_d(G_{ij},G_k) = 0 \Rightarrow (m_k - m_{ij}) > 3.(\sigma_k + \sigma_{ij})$$
$$I_d(G_{ij},G_k) > 0 \Rightarrow (m_k - m_{ij}) < 3.(\sigma_k + \sigma_{ij})$$

In other words, if $I_d(G_{ij},G_k)$ is a function of respectively the parameters and $(m_k - m_{ij})$ and $3.(\sigma_k + \sigma_{ij})$, then $I_d(G_{ij},G_k)$ is non-increasing with respect to its first argument and non-decreasing with respect to the second argument. Note that since the parameters are normalized in the interval [0,1], we propose the following representation for I_d:

$$I_d(G_{ij},G_k)=\begin{cases}\left|m_{ij}-m_k\right|\cdot exp\left(-3\cdot\left(\sigma_{ij}+\sigma_k\right)\right) & \text{si} \quad \sigma_{ij}\geq\sigma_k \\ 0 & \text{si} \quad \sigma_{ij}\prec\sigma_k\end{cases} \quad (5)$$

It's easily checked that $I_d(G_{ij},G_k)$ is non-increasing with respect to $(m_k - m_{ij})$, one may determine its derivative with respect to $(m_k - m_{ij})$.

3.2.2 Formulation of the proposed Algorithm

Let $P_{ij} = [m_{ij}, \sigma_{ij}]$ be the MFs parameters associated to input variable x_i obtained after the parameter learning phase. The aim of the clustering algorithm is to determine an optimal clusters set $\{v_k\}$, where $v_k = [m_k, \sigma_k]$, in order to replace the old fuzzy partition $\{P_{ij}\}$ by the new one $\{v_k\}$ according to the minimization of the following objective function [4][7]:

$$J_i^{(fcm)} = \sum_{i=1}^{n}\sum_{k=1}^{c} (\mu_{ki})^m . [d(P_{ij},v_k)]^2 \quad (6)$$

With respect to the constraint: $\sum_{k=1}^{c} \mu_{ki} = 1, \forall i=1, ., n$

The distance $d(P_{ij},v_k)$ between P_{ij} and the prototype v_k is defined by :

$$d(P_{ij},v_k)^2 = (P_{ij} - v_k)^t.A_j.(P_{ij} - v_k) \quad (7)$$

Where A_j is semi-defined positive matrix.

In other words, this algorithm is to find membership degree μ_{ki} of each variable P_{ij} to the center of the optimized class v_k [4], where M, c and m are respectively the number of data i=(1,.., M), the number of clusters initialized randomly and an arbitrary chosen scalar.

The new clustering algorithm supposes that the distance concept can be considered as an inclusion concept [9][12]. It can be formulated as follows:

$$\sum_{k=1}^{c} [d(P_{ij}, v_k)]^2 = \sum_{k=1}^{c} [I_d(G_{ij},G_k)]^2 \quad (8)$$

$\forall\, i\in[1,n]$ and $j\in[1,M]$, from (6) and (7), we have :

$$\sum_{k=1}^{c}\left(P_{ij}-v_k\right)^T A_j\left(P_{ij}-v_k\right)=\sum_{k=1}^{c}\left|m_{ij}-m_k\right|^2 \exp\left(-6(\sigma_{ij}+\sigma_k)\right) \quad (9)$$

Let us consider a new representation of the Gaussian parameters, so that, $P_{ij}^{(1)} = m_{ij}$, $P_{ij}^{(2)} = \sigma_{ij}$, $v_k^{(1)} = m_{ij}$ and $v_k^{(2)} = \sigma_{ij}$, and a new quadratic formulation of equation (9). Let us denote by B_1 and B_2 two matrix, where $B_1=\begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}$ and $B_2=\begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}$.

Then, Eq.8 can be formulated as follows:

$$\sum_{k=1}^{c} (P_{ij}-v_k)^t A_j(P_{ij}-v_k) - \sum_{k=1}^{c} |P_{ij}^{(1)}-v_k^{(1)}|^t.B_1.|P_{ij}^{(1)}-v_k^{(1)}| .exp(-6.(P_{ij}^{(2)}+v_k^{(2)})^t.B_2) = 0 \quad (10)$$

Leading to an equality constraint to the optimization problem, so the non-linear optimization problem (P_r) can be written as:

$$P_r: \begin{cases}\text{Minimize}\, J_i^{(fcm)}(\mu_{ki}, A_j, v_k), \text{under constraint} \\ g_r(\mu_{ki}, A_j, v_k) = 0, \forall r \in I = (1,..,n)\end{cases} \quad (11)$$

In our case, two constraints are considered (r=1,2):

$$g_1(\mu_{ki}) : \sum_{k=1}^{c} \mu_{ki} - 1 = 0 , \forall i=1, 2, .., n \quad (12)$$

$$g_2(\mu_{ki},A_j,v_k): \sum_{k=1}^{c}(P_{ij}-v_k)^t A_j(P_{ij}-v_k) - \sum_{k=1}^{c}|P_{ij}^{(1)}-v_k^{(1)}|^t B_1|P_{ij}^{(1)}-v_k^{(1)}|.exp(-6(P_{ij}^{(2)}+v_k^{(2)})^t B_2)=0 \quad (13)$$

Using a dual methods, the (P_r) is transformed into a min-max problem by introducing the Lagrange parameter vector λ_r, so that:

$$L(\mu_{ki},A_j,v_k,\lambda) = J_i^{(fcm)}(\mu_{ki},A_j,v_k) + \sum_{r=1}^{2} \lambda_r^{T}.g_r(\mu_{ki},A_j,v_k) \quad (14)$$

The substitution of (6), (12), (13) and (14) gives:

$$L(\mu_{ki},A_j,v_k,\lambda_1,\lambda_2) = \sum_{i=1}^{n}\sum_{k=1}^{c}[(\mu_{ki})^m.(P_{ij}-v_k)^t.A_j.(P_{ij}-v_k)] + \lambda_1.g_1(\mu_{ki}) + \lambda_2.g_2(\mu_{ki},A_j,v_k) \quad (15)$$

The optimal parameters are obtained by setting the derivative of Lagrangian according to each of its parameters $P_l = [\mu_{ki}, A_j, v_k, \lambda_1, \lambda_2]$ to zero, so that:

$$\frac{\partial L(\mu_{ki},A_j,v_k,\lambda_1,\lambda_2)}{\partial P_l} = 0 \quad (16)$$

The detail of the calculus is in [12]. Using the classical problem of fuzzy C-means clustering when the distance d is replaced by the matrix formulation [4], and assuming $m>1$, this leads us to:

$$\mu_{ki} = \frac{1}{\left(P_{ij}-v_k\right)^t A_j\left(P_{ij}-v_k\right)^{\frac{2}{m-1}}.\sum_{i=1}^{n}\frac{1}{\left(P_{ij}-v_k\right)^t A_j\left(P_{ij}-v_k\right)^{\frac{2}{m-1}}}} \quad (17)$$

Using the method of the gravity centers, the clusters centers coordinates v_k are computed as follows:

$$v_k = \frac{\sum_{i=1}^{n}(\mu_{ki})^m.P_{ij}}{\sum_{i=1}^{n}(\mu_{ki})^m} \quad (18)$$

The matrix A_j is computed as:

$$A_j = \frac{\sum_{k=1}^{c}\left|P_{ij}^{(1)}-v_k^{(1)}\right|^t B_1\left|P_{ij}^{(1)}-v_k^{(1)}\right|.exp\left(-6\left(P_{ij}^{(2)}-v_k^{(2)}\right)\right)^t B_2}{\sum_{k=1}^{c}\left(P_{ij}-v_k\right)\left(P_{ij}-v_k\right)} \quad (19)$$

The parameters λ_1 and λ_2 are computed as:

$$\lambda_1 = 1-(m.\sum_{i=1}^{n}\sum_{k=1}^{c}(\mu_{ki})^{m-1}(P_{ij}-v_k)^t A_j(P_{ij}-v_k)+\sum_{k=1}^{c}\mu_{ki}^{m}) \quad (20)$$

$$\lambda_2 = \frac{\sum_{i=1}^{n}\sum_{k=1}^{c}(\mu_{ki})^m\left(-A_j\left(P_{ij}-v_k\right)-\left(P_{ij}-v_k\right)^t A_j\right)}{\sum_{k=1}^{c}\left(W_{ki}(P_{ij},v_k)-V_{ki}(P_{ij},v_k,A_j)\right)} \quad (21)$$

The proposed fuzzy clustering algorithm can be summarized by the following steps:

1- *Initialization*

a- Random initialization of the clusters number c, their centers v_k.

b- Introduce the constants parameters: $m = 2$, $\alpha_1 = 1.05$ et $\alpha_2 = 0.95$.

c- Introduce all the data P_{ij} for the treatment.

2- *Loop*

d- Compute the matrix A_j from (18).

e- Generate a new partition μ_{ki} from (17).

f- Compute the criteria $J_i^{(fcm)}$, so that :

$$J_i^{(fcm)} = \sum_{i=1}^{n}\sum_{k=1}^{c}((\mu_{ki})^m.(P_{ij}-v_k)^t.A_j.(P_{ij}-v_k))$$

g- Test1: $\begin{cases}\text{If } J_i^{(fcm)} \leq Minimum \text{ then go to step i-}\\ \text{else go to step h-}\end{cases}$

h- Determine a new clusters centers v_k:

$$v_k = \frac{\sum_{i=1}^{n}(\mu_{ki})^m.P_{ij}}{\sum_{i=1}^{n}(\mu_{ki})^m} \text{ and go to step d-}$$

i- Verify the rule-base property by computing:

$$\text{Test-2: } \begin{cases} if \begin{cases}\sum_{k=1}^{c}\mu_{ki} > \alpha_1 \Rightarrow c = c-1 \\ \sum_{k=1}^{c}\mu_{ki} < \alpha_2 \Rightarrow c = c+1\end{cases} \text{ and go to step d-} \\ \text{else STOP}\end{cases}$$

3- *Stop* (End of loop)

3.3 Real Time Adaptation

In order to ensure an efficient real time force position control and to compensate the fuzzy sets fusion effects, the NFC is adapted on line by tuning the consequences parameters. The real time adaptation algorithm is given:

$$w(p+1) = w(p) - \eta_w \frac{\partial J(p)}{\partial w(p)} \quad (22)$$

Where η_w represents the convergence speed parameter and J the cost function (**Fig.1**) given by:

$$J(p) = \frac{1}{2}\sum_{k=1}^{p}(Fd(k)-Fr(k))^2 \quad (23)$$

The derivatives $\frac{\partial J(p)}{\partial w(p)}$ can be formulated by:

$$\frac{\partial J(p)}{\partial w(p)} = \frac{\partial J(p)}{\partial F_r(p)}\cdot\frac{\partial F_r(p)}{\partial w(p)} = -\sum_{k=1}^{p}\left(F_d(k)-F_r(k)\right)\cdot\frac{\partial F_r(p)}{\partial u_{fl}(p)}\cdot\frac{\partial u_{fl}(p)}{\partial w(p)} \quad (24)$$

Since $\frac{\partial F_r(p)}{\partial u_{fl}(p)}$ defines the sensitivity matrix of the system, which is known, only $\frac{\partial u_{fl}(p)}{\partial w(p)}$ must be determined. One can note that if the parameters which determine the convergence speed are correctly chosen, all the fluctuations which act on the system are compensated and leading to the desired behavior.

4. Simulation Results

In order to demonstrate the validity of our approach, the control structure has been implemented on a planar two-link manipulator with two revolute joints. The robot has to follow a circular trajectory under contact force constraints. The NFC is divided into two sub-systems. The controller input is the operational space force error vector $\Delta F_i(\Delta F_x, \Delta F_y)$, and the output is the vector $u_d(u_{dx}, u_{dy})$ consisting of the operational space trajectory perturbation which is added to the operational space desired trajectory for position control loop (Fig.1). The set of MFs associated to each input variable, is randomly initialed. In the first phase, parameter learning is ensured and the identification results are shown on Fig.6.

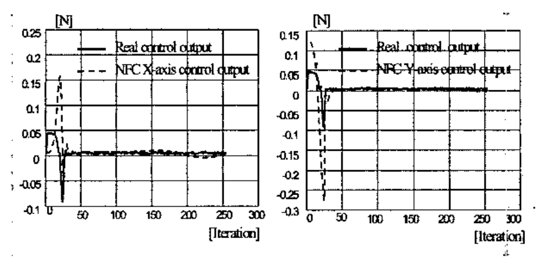

Fig.6 NFC Outputs and Desired Outputs in Task Space

Fig.7 illustrates the convergence behavior of each NFC sub-system. The sum-squared error is decreasing with epochs to a value 0.037 for the first NFC sub-system and 0.023 for the second one.

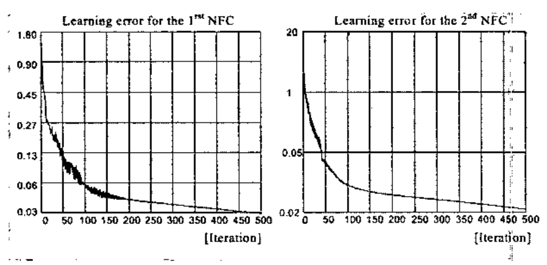

Fig.7 Learning Error for the Two NFC

In the second phase, the clustering algorithm is used. Fig.8 represents the contribution of the input variables in the overall fuzzy rules after the learning parameters phase in each NFC sub-system. Due to the MFs overlapping, the fuzzy clustering algorithm is used to reduce the MFs number and optimize the fuzzy base rule. The discontinued lines represent the generated classes after clustering

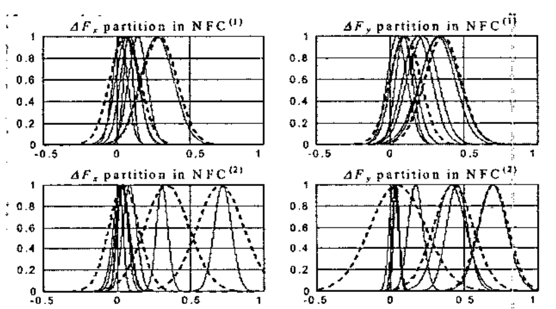

Fig.8 Optimal Clusters and Initials MFs

Fig.9 gives optimal clusters within initial MFs obtained after the application of the fuzzy clustering algorithm, where (•) represents the cluster center and (+) represents the optimized datum.

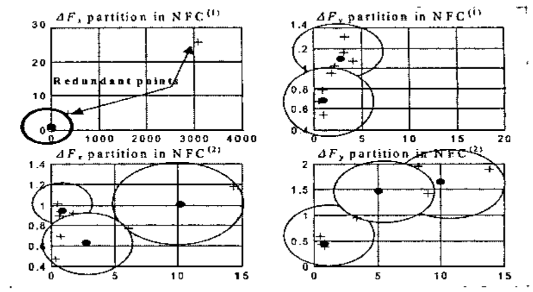

Fig.9 Optimal Clusters with 2D Representation

From Fig.9, there exit two redundant points in the numerous optimizing data, which can be regarded as an inconsistency with the trends of the NFC. These points are removed from the original data, because there does not exist a cluster center around any redundant datum. The circles define the clusters. They include fuzzy sub-sets, which have the highest membership degrees to the considered clusters centers. The learning error for each NFC sub-system is shown on Fig.10.

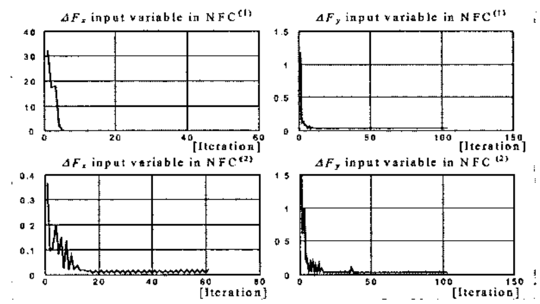

Fig.10 Learning Error During Clustering Phase

Table.1 shows the performances of the proposed learning algorithm before and after classification. At the opposite to conventional fuzzy clustering algorithm based on class of methods developed by Bezdek, the proposed algorithm doesn't require any initial knowledge on clusters number to be identified and on the distribution of all the optimizing datum only, but also the fuzzy rule-base consistency is respected.

Fuzzy sets number	Before learning		32
	After learning		10
Number of rules	Before learning		16
	After learning		05
Convergence behavior	$NFC^{(1)}$	ΔF_x	0.0024
		ΔF_y	0.0040
	$NFC^{(2)}$	ΔF_x	0.0250
		ΔF_y	0.0012
Convergence Error in term of number of epochs	$NFC^{(1)}$	ΔF_x	5
		ΔF_y	10
	$NFC^{(2)}$	ΔF_x	17
		ΔF_y	12
Initial knowledge			**Not required**
Parameters initialization (clusters number, centers, ..)			**Randomly**
Rule-base completude property			**Respected**

Tab.1 Classification Phase Obtained Results

The application of the on line adaptation algorithm leads to results given on Fig.11 (the measured forces in task space) and Fig.12 (the robot end-effector trajectory).

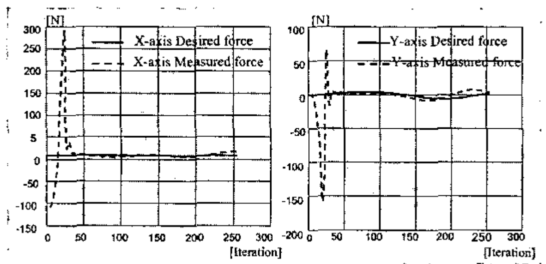

Fig.11 Measured and Desired Forces in Task Space

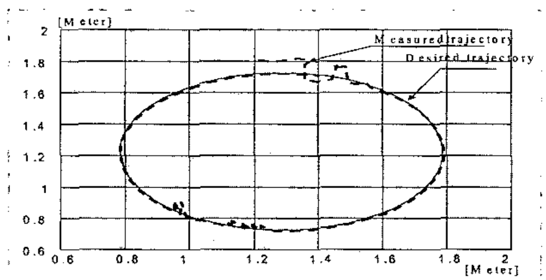

Fig.12 Robot End- effector Trajectory

Fig.13 shows the temporal behavior of the criteria. It exhibits, a peak with a value 700 during the contact at the epoch 20 and stabilizes around 0.087, at 55 epochs.

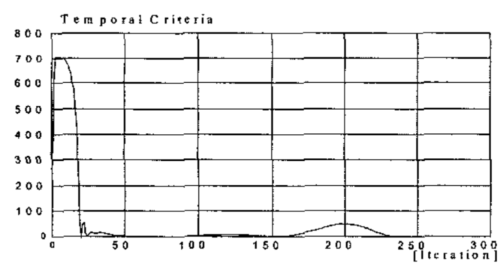

Fig.13 Temporal Criteria Behavior

5. Conclusion

In this paper, an adaptive neuro-fuzzy approach to solve the control of constrained dynamic systems is proposed. Such approach is very indicated for complex systems where classical approaches failed. The designing methodology of a new NFC dedicated to constrained tasks has been presented. Compared to neural networks control and fuzzy control, taken separately, the proposed approach has some advantages over them. The architecture allows the meaning of all the internal parameters under the form of the fuzzy rule base. A new fuzzy clustering algorithm based on inclusion concept that allows simplification of the NFC parameters is presented. In order to show the validity of this approach, a task consisting in a trajectory following under environment constraints executed by a planar manipulator has been presented. The analysis and evaluation of the results show the suitability and the efficiency of the proposed approach.

References

[1] P.J. Antsaklis, " A brief introduction to the theory and applications of hybrid systems ", IEEE, Special Issue on Hybrid Systems: Theory and Applications, vol.88, n°.7, July 2000, pp. 879-887.

[2] J.C. Bezdek, C. Coray and R. Gunderson, " Detection and characterization of cluster structure ", Part.1, SIAM J. Appl. Math., 40 (2), 1981, pp. 339-357.

[3] K.M. Berger and K.A. Loparo, " A hierarchical framework for learning control ", Center for Automation and Intelligent Systems, Case Western Reserve Univ., May 1988.

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

[5] R.N. Dave, " Fuzzy shell-clustering and applications to circle detection in digital images ", Int. J. of General Systems, vol.16, 1990, pp. 343-355.

[6] D.E. Gustafson and J. Keller, " A possibilistic approach to clustering ", IEEE Tran. on Fuzzy Systems, vol.1, 1993, pp. 98-109.

[7] H. Imai and A. Tanaka, " A method of identifying influential data in fuzzy clustering ", IEEE Trans. on Fuzzy Systems, vol. 6, n°.1, Feb.98, pp. 90-101.

[8] J.S.R. Jang, " Neuro-fuzzy modeling: architecture, analyses and applications ", Ph.D. Thesis, Dep. of Electrical Engineering and Computer Science, Univ. of California, Berkeley USA, July 1993.

[9] S. Nefti, K. Djouani and J. Pontnau, " Clustering and adaptive mobile robot navigation in a moving and unknown environment ", Pro. INES'98-IEEE, Vienna, Austria, 1998, pp. 55-60.

[10] Z. Pingan and L. Renhou, " A new approach to fuzzy identification for complex ", IEEE Int. Conf. on Fuzzy Systems, 1996, pp. 1309-1313.

[11] T. Takagi and M. Sugeno, " Fuzzy identification of systems and its applications to modelling and control ", IEEE Trans. on Systems, Man and Cybernetics, SMC 15(1), 1987, pp. 116-132.

[12] Y. Touati, K. Djouani and Y. Amirat, " Adaptive fuzzy logic controller for robots manipulators ", IEEE, The 10th Artificial Neural Networks in Engineering Conference ANNIE 2001, St Louis, Missouri USA, vol.11, Nov.2001, pp. 247-254.

[13] L.X. Wang and J.M. Mendel, " Generating fuzzy rules by learning from examples ", IEEE Trans. on Systems, Man and Cybernetics, vol.22, n°.6, Nov.1992, pp. 1414-1427.