New Trends in Neuro-Fuzzy Adaptive Control Systems

Manolis Christodoulou, *Senior Member, IEEE,* and Yiannis Boutalis, *Member, IEEE*

***Abstract*—We use a new definition of Neuro-Fuzzy Dynamical Systems, using the concept of Fuzzy Dynamical Systems (FDS) in conjunction with High Order Neural Network Functions (F-HONNFs). The dynamical System is assumed nonlinear and totally unknown. We first propose its approximation by a special form of a fuzzy dynamical system (FDS) and in the sequel the fuzzy rules are approximated by appropriate HONNF's. Thus the identification scheme leads to a Recurrent High Order Neural Network, which however, takes into account the fuzzy output partitions of the initial FDS. The proposed scheme does not require a priori experts' information on the number and type of input variable membership functions, making it less vulnerable to initial design assumptions.**

After the identification process we adaptively control the system indirectly. By doing so, we present weight updating laws for the involved HONNs. With rigorous proofs we guarantee that the errors converge to zero exponentially fast, or at least become uniformly ultimately bounded. At the same time we guarantee stability by proving that all signals in the closed loop remain bounded.

During both the identification and control process we assume, first that we know the centers and shapes of membership functions, and we identify the HONN parameters in which case we get a directional variation. Thus in order to guarantee existence of the control law, we define a new method replacing the well known projection, which is termed parameter hopping and thus we rigorously prove existence of the control law, guaranteeing stability properties.

Simulations illustrate the potency of the method and comparisons with conventional approaches are given. The simulation tests are based on benchmark examples. Also, the applicability of the method is tested on a DC Motor system where it is shown that by following the proposed procedure one can obtain asymptotic regulation.

***Index Terms*—Neural Networks, Fuzzy Systems, Adaptive Control, Parameter Hopping.**

I. INTRODUCTION

NONLINEAR time invariant dynamical systems can be represented by general nonlinear dynamical equations of the form

$$\dot{x} = f(x, u) \tag{1}$$

The mathematical description of the system is required, so that we are able to control it. Unfortunately, the exact mathematical model of the plant, especially when this is highly nonlinear and complex, is rarely known and thus appropriate identification schemes have to be applied which will provide us with an approximate model of the plant.

It has been established that neural networks and fuzzy inference systems are universal approximators [1], [2], [3],i.e., they can approximate any nonlinear function to any prescribed accuracy provided that sufficient hidden neurons and training data or fuzzy rules are available. Recently, the combination of these two different technologies has given rise to fuzzy neural or neuro fuzzy approaches, that are intended to capture the advantages of both fuzzy logic and neural networks. Numerous works have shown the viability of this approach for system modeling [4] - [12].

The neural and fuzzy approaches are most of the time equivalent, differing between each other mainly in the structure of the approximator chosen. Indeed, in order to bridge the gap between the neural and fuzzy approaches several researchers introduce adaptive schemes using a class of parameterized functions that include both neural networks and fuzzy systems [6] - [12]. Regarding the approximator structure, linear in the parameters approximators are used in [10], [13], and nonlinear in [14], [15], [16].

Adaptive control theory has been an active area of research over the past years [13]-[51]. In the neuro or neuro fuzzy adaptive control two main approaches are followed. In the indirect adaptive control schemes [13] - [19], first the dynamics of the system are identified and then a control input is generated according to the certainty equivalence principle. In the direct adaptive control schemes [20] - [25] the controller is directly estimated and the control input is generated to guarantee stability without knowledge of the system dynamics. Also, many researchers focus on robust adaptive control that guarantees signal boundedness in the presence of modeling errors and bounded disturbances [26]. In [27] both direct and indirect approaches are presented, while in [28],[29] a combined direct and indirect control scheme is used.

Recently [41], [42], high order neural network function approximators (HONNFs) have been proposed for the identification of nonlinear dynamical systems of the form (1), approximated by a Fuzzy Dynamical System. This approximation depends on the fact that fuzzy rules could be identified with the help of HONNFs.

In this paper HONNFs are also used for the neuro fuzzy indirect control of nonlinear dynamical systems, which comprises of two interrelated phases: first the identification of the model and second the control of the plant.

The identification phase usually consists of two categories: structure identification and parameter identification. Structure identification involves finding the main input variables out of

M.A Christodoulou is with Department of Electronic and Computer Engineering, Technical University of Crete, Chania, Crete, GREECE 73100 GR e-mail: manolis@ece.tuc.gr

Y. Boutalis is with the Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, GREECE, 67100 GR and with the Department of Electrical, Electronic and Communication Engineering, Chair of Automatic Control, University of Erlangen-Nuremberg, GERMANY e-mail: ybout@ee.duth.gr;

all possible, specifying the membership functions, the partition of the input space and determining the number of fuzzy rules which is often based on a substantial amount of heuristic observation to express proper strategy's knowledge. Most of structure identification methods are based on data clustering, such as fuzzy C-means clustering [9], mountain clustering [11], and subtractive clustering [12]. These approaches require that all input-output data are ready before we start to identify the plant. So these structure identification approaches are off-line.

In the proposed approach structure identification is also made off-line based either on human expertise or on gathered data. However, the required a-priori information obtained by linguistic information or data is very limited. The only required information is an estimate of the centers of the output fuzzy membership functions. Information on the input variable membership functions and on the underlying fuzzy rules is not necessary because this is automatically estimated by the HONNFs. This way the proposed method is less vulnerable to initial design assumptions. The parameter identification is then easily addressed by HONNFs, based on the linguistic information regarding the structural identification of the output part and from the numerical data obtained from the actual system to be modeled.

We consider that the nonlinear system is affine in the control and could be approximated by a dynamic model using two independent adaptive fuzzy subsystems. Every fuzzy subsystem is approximated by a family of HONNFs, each one being related with a group of fuzzy rules, every group associated with one center of an output fuzzy membership function. This way, the identification scheme leads up to a Recurrent High Order Neural Network, which however takes into account the fuzzy output partitions of the initial FDS. Weight updating laws are given and we prove that when the structural identification is appropriate then the error reaches zero very fast. Also, an appropriate state feedback is constructed to achieve asymptotic regulation of the output, while keeping bounded all signals in the closed loop. The existence of the control signal is always assured by introducing a novel method of parameter hopping, which is incorporated in the weight updating law.

The paper is organized as follows. Section II presents preliminaries related to the concept of adaptive fuzzy systems (AFS) and the terminology used in the remaining paper, while Section III reports on the ability of HONNFs to act as fuzzy rule approximators. The indirect neuro fuzzy regulation of affine in the control dynamical systems is presented in Section IV, where the method of parameter hopping is explained and the associated weight adaptation laws are given. Simulation results on the identification of well known benchmark problems are given in Section V and the performance of the proposed scheme is compared to other well known approaches of the literature. Also, simulation results on the control of a DC Motor system are given, showing that by following the proposed procedure one can obtain asymptotic regulation. Finally, Section VI concludes the work.

II. PRELIMINARIES

In this section we briefly present the notion of adaptive fuzzy systems and their conventional representation. We are also introducing the representation of fuzzy systems using the fuzzy rule indicator functions, which is used for the development of the proposed method.

A. Adaptive Fuzzy Systems

The performance, complexity, and adaptive law of an adaptive fuzzy system representation can be quite different depending upon the type of the fuzzy system (Mamdani or Takagi-Sugeno). It also depends upon whether the representations is linear or nonlinear in its adjustable parameters. Adaptive fuzzy controllers depend also on the type of the adaptive fuzzy subsystems they use. Suppose that the adaptive fuzzy system is intended to approximate the nonlinear function $f(x)$. In the mamdani type, linear in the parameters form, the following fuzzy logic representation is used [2],[3]:

$$f(x) = \sum_{l=1}^{M} \theta_l \xi_l(x) = \theta^T \xi(x) \tag{2}$$

where M is the number of fuzzy rules, $\theta = (\theta_1, ..., \theta_M)^T$, $\xi(x) = (\xi_1(x), ..., \xi_M(x))^T$ and $\xi_l(x)$ is the fuzzy basis function defined by

$$\xi_l(x) = \frac{\prod_{i=1}^{n} \mu_{F_i^l}(x_i)}{\sum_{l=1}^{M} \prod_{i=1}^{n} \mu_{F_i^l}(x_i)} \tag{3}$$

θ_l are adjustable parameters, and $\mu_{F_i^l}$ are given membership functions of the input variables (can be Gaussian, triangular, or any other type of membership functions).

In Tagaki-Sugeno formulation $f(x)$ is given by

$$f(x) = \sum_{l=1}^{M} g_l(x) \xi_l(x) \tag{4}$$

where $g_l(x) = a_{l,0} + a_{l,1}x_1 + \ldots + a_{l,n}x_n$, with $x_i, i = 1 \ldots n$ being the elements of vector x and $\xi_l(x)$ being defined in (3). According to [3], (4) can also be written in the linear to the parameters form, where the adjustable parameters are all $a_{l,i}, l = 1 \ldots M, i = 1 \ldots n$.

From the above definitions it is apparent in both, Mamdani and Tagaki-Sugeno forms that the success of the adaptive fuzzy system representations in approximating the nonlinear function $f(x)$ depends on the careful selection of the fuzzy partitions of input and output variables. Also, the selected type of the membership functions and the proper number of fuzzy rules contribute to the success of the adaptive fuzzy system. This way, any adaptive fuzzy or neuro-fuzzy approach, following a linear in the adjustable parameters formulation becomes vulnerable to initial design assumptions related to the fuzzy partitions and the membership functions chosen. In this paper this drawback is largely overcome by using the concept of rule indicator functions, which are in the sequel approximated by High order Neural Network function approximators (HONNFs). This way there is not any need for initial design assumptions related to the membership values and the fuzzy partitions of the if part.

B. Fuzzy system description using rule indicator functions

Let us consider the system with input space $u \subset R^m$ and state - space $x \subset R^n$, with its i/o relation being governed by the following equation

$$z^t = f(x^t, u^t) \tag{5}$$

where $f(\cdot)$ is a continuous function and the superscript t denotes the temporal variable. In case the system is dynamic the above equation could be replaced by the following difference equation

$$x^{t+1} = f(x^t, u^t) \tag{6}$$

where the superscript t denotes the temporal variable, $t = 1, 2, ...$

By setting $y = [x, u]$ and omitting superscript t, Eq. (5) may be rewritten as follows

$$z = f(y) \tag{7}$$

In many practical situations, we are unable to measure accurately the states and inputs of a system of the form in (5); in most cases, we are provided with cheap sensors, expert's opinions, e.t.c which provide us with imprecise estimations of the state and input vectors. Thus, instead of vectors *x* and *u* we are provided with some linguistic variables $\tilde{x}_i$ and $\tilde{u}_i$, respectively.

Let now $\tilde{y} := (\tilde{x}, \tilde{u})$ and suppose that each linguistic variable $\tilde{y}_i$ belongs to a finite set L_i with cardinality k_i, i.e. $\tilde{y}_i$ takes one of k_i variables. Let also $\tilde{y}_{ij}$ denotes the *ith* element of the set L_i. Then we may define a function $\tilde{h}_i : R \rightarrow L_i$ to be the output function of the system in Eq. (7) in the case that

$$\tilde{y}_i = \tilde{h}_i(y_i) \tag{8}$$

Note that $\tilde{h}_i(\cdot)$ maps the real axis into a set of linguistic variables L_i , and thus $\tilde{h}_i(\cdot)$ is not defined in the usual way. In order to overcome such a problem we define the function $\tilde{h}_i : R \rightarrow \{1, 2, ..., k_i\}$ as follows

$$\tilde{h}_i(y_i) = \tilde{y}_{ij} \Longleftrightarrow h_i(y_i) = j \tag{9}$$

Since $h_i(\cdot)$ is very similar to $\tilde{h}_i(\cdot)$, we will call the function $h_i(\cdot)$ the *ith* output of the system in Eq. (7). Also, $\tilde{h}_i(\cdot)$ and consequently $h_i(\cdot)$ is related with the structural identification part mentioned in section I and arrive after using an automatic procedure based on system operation data or after consulting human experts advising on how to partition the system variables.

Following the standard approach in fuzzy systems theory we associate with each $\tilde{y}_{ij}$ a membership function $\tilde{\mu}_{ij}(y_i) \in [0, 1]$ which satisfies

$$\tilde{\mu}_{ij}(y_i) = \max_l \tilde{\mu}_{il}(y_i) \Longleftrightarrow h_i(y_i) = j \tag{10}$$

From the definition of the functions $\tilde{h}_i(\cdot)$ [or $h_i(\cdot)$] we have that the space $\mathcal{Y} = \mathcal{X} \times \mathcal{U}$ is partitioned in the following way: let $\mathcal{Y}_{ij}$ be defined as follows

$$\mathcal{Y}_{ij} = \{y_i \in R : h_i(y_i) = j\} \tag{11}$$

i.e. $\mathcal{Y}_{ij}$ denotes the set of all the variables y_i that output the same linguistic variable $\tilde{y}_{ij}$. Thus y is partitioned into disjoint subsets $\mathcal{Y}_{j_1,j_2,...,j_{n+m}}$ defined as follows

$$\mathcal{Y}_{j_1,j_2,...,j_{n+m}} := \mathcal{Y}_{1j_1} \times \cdots \times \mathcal{Y}_{(n+m)j_{n+m}}, j_i \in \{1, 2, ..., k_i\} \tag{12}$$

In a similar way we may define the sets $\mathcal{X}_{ij}$, $\mathcal{U}_{ij}$, $\mathcal{Z}_{ij}$ and the sets $\mathcal{X}_{j_1,j_2,...,j_n}$, $\mathcal{U}_{j_1,j_2,...,j_n}$ and $\mathcal{Z}_{j_1,j_2,...,j_n}$. Note now the following fact: for two vectors $(x^{(1)}, u^{(1)}) \in \mathcal{Y}_{j_1,j_2,...,j_{n+m}}$ and $(x^{(2)}, u^{(2)}) \in \mathcal{Y}_{j_1,j_2,...,j_{n+m}}$ there maybe

$$h_i(f_i(x^{(1)}, u^{(1)})) \neq h_i(f_i(x^{(2)}, u^{(2)})) \tag{13}$$

for some $i \in \{1, 2, ..., n\}$, i.e. two input vectors belonging to the same subset $\mathcal{Y}_{j_1,j_2,...,j_{n+m}}$ may point - through the vector - field $f(\cdot)$, to different subsets $\mathcal{Z}_{l_1,l_2,...,l_n}$. Let now $\Omega^{l_1,l_2,...,l_n}_{j_1,j_2,...,j_{n+m}}$ be defined as the subset of $\mathcal{Y}_{j_1,j_2,...,j_{n+m}}$ that points - through the vector - field $f(\cdot)$, to the subsets $\mathcal{Z}_{l_1,l_2,...,l_n}$, i.e

$$\Omega^{l_1,l_2,...,l_n}_{j_1,j_2,...,j_{n+m}} :=$$

$$= \{(x, u) \in \mathcal{Y}_{j_1,j_2,...,j_{n+m}} : h_1(z_1) = l_1, ..., h_n(z_n) = l_n\}$$

and define the transition possibilities $\pi^{l_1,...,l_n}_{j_1,...,j_{n+m}}$ as follows

$$\pi^{l_1,...,l_n}_{j_1,...,j_{n+m}} := \frac{\int_{(x,u)\in\Omega^{l_1,...,l_n}_{j_1,...,j_{n+m}}} dXdU}{\int_{(x,u)\in y_{j_1,...,j_{n+m}}} dXdU} \tag{14}$$

where $\pi^{l_1,...,l_n}_{j_1,...,j_{n+m}}$ is a number belonging to a set [0,1] that represents the fraction of the vectors (x,u) in $y_{j_1,...,j_{n+m}}$ that points - through the vector field $f(\cdot)$ to the set $\chi_{l_1,...,l_n}$. Obviously

$$\sum_{l_1,...,l_n} \pi^{l_1,...,l_n}_{j_1,...,j_{n+m}} = 1 \tag{15}$$

In order to present the lemma of Section III, we define the indicator function: Let $I^{l_1,l_2,...,l_n}_{j_1,j_2,...,j_{n+m}}$ denote the indicator function of the subset $\Omega^{l_1,l_2,...,l_n}_{j_1,j_2,...,j_{n+m}}$, that is,

$$I^{l_1,...,l_n}_{j_1,...,j_{n+m}}(x, u) = \begin{cases} \alpha & if\ (x, u) \in \Omega^{l_1,...,l_n}_{j_1,...,j_{n+m}} \\ 0 & otherwise \end{cases} \tag{16}$$

where α denotes the firing strength of the rule.

Using the above definitions, we can see that the system in Eq. (7) is described by fuzzy rules of the form

$$R^{l_1,...,l_n}_{j_1,...,j_{n+m}} \Leftrightarrow \left\{ \begin{array}{c} IF\ y_1\ is\ \tilde{y}_{1j_1}\ AND... \\ AND\ y_{n+m}\ is\ \tilde{y}_{(n+m)j_{n+m}} \\ THEN \\ z_1\ is\ \tilde{z}_{1l_1}\ AND...AND\ z_n\ is\ \tilde{z}_{nl_n} \\ with\ possibility\ \pi^{l_1,...,l_n}_{j_1,...,j_{n+m}} \end{array} \right. \tag{17}$$

where obviously $\tilde{y}_{ij_i} = \tilde{h}_i(y^t_i)$ and $\tilde{z}_{il_i} = \tilde{h}_i(z_i) = \tilde{h}_i(f_i(x, u))$.

In the above notation, if $j_1 = l_1$, $j_2 = l_2$ and ... and $j_n = l_n$, then these points participate to the definition of the same fuzzy rule. If $j_1 \neq l_1$ or $j_2 \neq l_2$ or or $j_n \neq l_n$, then these points define alternative fuzzy rules describing this transition. Consider now the next definition.

Definition 1 (FS): A Fuzzy System - (FS) is a set of Fuzzy Rules of the form $(R^{l_1,l_2,\ldots,l_n}_{j_1,j_2,\ldots,j_{n+m}})$; the system in Eq. (5) is called the Underlying System - (US) of the previously defined FS. Alternatively, the system in Eq. (5) will be called a Generator of the FS that is described by the rules $(R^{l_1,l_2,\ldots,l_n}_{j_1,j_2,\ldots,j_{n+m}})$.

Due to the linguistic description of the variables of the FS it is not rare to have more than one systems of the form in Eq. (7) to be generators for the FS that is described by the rules $(R^{l_1,l_2,\ldots,l_n}_{j_1,j_2,\ldots,j_{n+m}})$.

Define now the following system

$$z = \sum \bar{z}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} \times I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(\chi, u) \tag{18}$$

Where $\bar{z}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} \in R^n$ be any vector satisfying $h_i(\bar{z}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(i)) = l_i$ where $\bar{z}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(i)$ denotes the i^{th} entry of $\bar{z}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}$ Then, according to [41], [42] the system in (18) is a generator for the FS $(R^{l_1,l_2,\ldots,l_n}_{j_1,j_2,\ldots,j_{n+m}})$.

It is obvious that Eq. (18) can be also valid for dynamic systems. In its dynamical form it becomes

$$\chi^{t+1} = \sum \bar{x}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} \times I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(\chi^t, u^t) \tag{19}$$

Where $\bar{x}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} \in R^n$ be any vector satisfying $h_i(\bar{x}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(i)) = l_i$ where $\bar{x}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(i)$ denotes the i^{th} entry of $\bar{x}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}$

III. THE HONNF'S AS FUZZY RULE APPROXIMATORS

The main idea in presenting the main result of this section lies on the fact that functions of high order neurons are capable of approximating discontinuous functions; thus, we use high order neural network functions in order to approximate the indicator functions $I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}$. However, in order the approximation problem to make sense the space $\mathcal{Y} := \mathcal{X} \times \mathcal{U}$ must be compact. Thus, our first assumption is the following:

(A.1) $\mathcal{Y} := \mathcal{X} \times \mathcal{U}$ is a compact set.

Notice that since $\mathcal{Y} \subset \Re^{n+m}$ the above assumption is identical to the assumption that it is closed and bounded. Also, it is noted that even if $\mathcal{Y}$ is not compact we may assume that there is a time instant T such that (x^t, u^t) remain in a compact subset of $\mathcal{Y}$ for all $t < T$; i.e. if $\mathcal{Y}_T := \{(x^t, u^t) \in \mathcal{Y}, t < T\}$ We may replace assumption (A.1) by the following assumption

(A.2) $\mathcal{Y}_T$ is a compact set.

It is worth noticing, that while assumption (A.1) requires the system in Eq. (6) solutions to be bounded for all $u^t \in U$ and $x^0 \in X$, assumption (A.2) requires the system in Eq. (6) solutions to be bounded for a finite time period; thus, assumption (A.1) requires the system in Eq. (6) to be BIBS stable while assumption (A.2) is valid for systems that are not BIBS stable and, even more, for unstable systems and systems with finite escape times.

We are now ready to show that high order neural network functions are capable of approximating the indicator functions $I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}$ Let us define the following high order neural network functions (HONNFs).

$$N(x, u; w, L) = \sum_{k=1}^{L} w_k \prod_{j \in I_k} \Phi_j^{d_j(k)} \tag{20}$$

Where $\{I_1, I_2, ..., I_L\}$ is a collection of L not-ordered subsets of $\{1, 2, ..., m+n\}$, $d_j(k)$ are non-negative integers, Φ_j are sigmoid functions of the state or the input, which are the elements of the following vector

$$\Phi = \begin{bmatrix} \Phi_1 \\ . \\ . \\ . \\ \Phi_n \\ \Phi_{n+1} \\ . \\ . \\ . \\ \Phi_{m+n} \end{bmatrix} = \begin{bmatrix} S(x_1) \\ . \\ . \\ . \\ S(x_n) \\ S(u_1) \\ . \\ . \\ . \\ S(u_m) \end{bmatrix} \tag{21}$$

where

$$S(u)\, or\, S(x) = a\frac{1}{1+e^{-\beta x}} - \gamma \tag{22}$$

and $w := [w_1 \cdots w_L]^T$ are the HONNF weights. Eq. (20) can also be written

$$N(x, u; w, L) = \sum_{k=1}^{L} w_k s_k(x, u) \tag{23}$$

Where $s_k(x, u)$ are high order terms of sigmoid functions of the state and/or input.

The next lemma [41] states that a HONNF of the form in Eq. (23) can approximate the indicator function $I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}$.

Lemma 1: Consider the indicator function $I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}$ and the family of the HONNFs $N(x, u; w, L)$. Then for any $\epsilon > 0$ there is a vector of weights $w^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}$ and a number of $L^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}$ high order connections such that

$$\sup_{(x,u)\in\bar{\mathcal{Y}}} \{I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(x,u) - \\ -N(x, u; w^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}, L^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n})\} < \varepsilon$$

where $\bar{\mathcal{Y}} \equiv \mathcal{Y}$ if assumption (A.1) is valid and $\bar{\mathcal{Y}}_T \equiv \mathcal{Y}$ if assumption (A.2) is valid.

Let us now keep $L^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}$ constant, i.e. let us preselect the number of high order connections, and let us define the optimal weights of the HONNF with

$L^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}$ high order connections as follows

$$\bar{w}^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n} := \arg \min_{w\in R^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}} \times$$

$$\left\{ \sup_{(x,u)\in\bar{\mathcal{Y}}} \left| I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(x,u) - N(x,u;w,L^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}) \right| \right\}$$

and the modeling error as follows

$$\nu^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(x,u) = I^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(x,u) -$$

$$-N(x,u;w^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}, L^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n})$$

It is worth noticing that from Lemma 1 we have that $\sup_{(x,u)\in\bar{y}} \left| \nu^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(x,u) \right|$ can be made arbitrarily small by simply selecting appropriately the number of high order connections.

Using the approximation Lemma 1 it is natural to approximate system in Eq. (19) by the following dynamical system

$$z^{t+1} = \sum \bar{x}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(x,u) \times$$

$$\times N(z^t, u^t; w^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n}, L^{j_1,\ldots,j_{n+m};l_1,\ldots,l_n})$$

Let now $\chi^t(\chi^0, u^t)$ denote the solution in Eq. (19) given that the initial state at $t = 0$ is equal to χ^0 and the input is u^t. Similarly we define $z^t(z^0, u^t)$. Also let

$$\nu(z^t, u^t) = \sum \left(\bar{x}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(x,u) \times \nu^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}(z^t, u^t) \right) \quad (24)$$

Then, it can be easily shown that

$$z^t(z^0, u^t) = \chi^t(z^0, u^t) + \nu(z^t, u^t) \quad (25)$$

Note now that from the approximation Lemma 1 and the definition of $\nu(z^t, u^t)$ we have that modeling error can be made arbitrarily small provided that (z^t, u^t) remain in a compact set (e.g. $\bar{y}$).

Theorem 1: [41],[42] Consider the FDS ($R^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}}$) and suppose that system in Eq. (6) is its underlying system. Assume that either assumptions (**A.1**) or (**A.2**) hold. Also consider the RHONN in [42]. Then, for any $\varepsilon > 0$ there exists a matrix Θ^* and a number L^* high order connections and $\Theta = \Theta^*$ is a generator for the FDS described by the rules

$$R^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} \Leftrightarrow \left\{ \begin{array}{c} IF\ y_1\ is\ \tilde{y}_{1j_1}\ AND... \\ AND\ y_{n+m}\ is\ \tilde{y}_{(n+m)j_{n+m}} \\ THEN \\ \chi_1\ is\ \tilde{y}_{1l_1}\ AND...AND\ \chi_n\ is\ \tilde{y}_{nl_n} \\ with\ possibility\ \widehat{\pi}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} \end{array} \right.$$

where

$$\max \left| \pi^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} - \widehat{\pi}^{l_1,\ldots,l_n}_{j_1,\ldots,j_{n+m}} \right| < \varepsilon$$

IV. Indirect adaptive neuro-fuzzy control

A. Neuro fuzzy representation and identification

We consider affine in the control, nonlinear dynamical systems of the form

$$\dot{x} = f(x) + G(x) \cdot u \quad (26)$$

where the state $x \in R^n$ is assumed to be completely measured, the control u is in R^n, f is an unknown smooth vector field called the drift term and G is a matrix with columns the unknown smooth controlled vector fields g_i, $i = 1, 2, ..., n$ and $G = [g_1, g_2, \ldots, g_n]$. The above class of continuous-time nonlinear systems are called affine, because in (26) the control input appears linear with respect to G. The main reason for considering this class of nonlinear systems is that most of the systems encountered in engineering, are by nature or design, affine. Furthermore, we note that non affine systems of the form given in (1) can be converted into affine, by passing the input through integrators, a procedure known as dynamic extension.

In our approach, referred to as *indirect adaptive fuzzy-HONNF control*, the plant parameters are estimated on-line except of the state fuzzy partitions, which are used to calculate the controller parameters. The basic structure of the *indirect adaptive fuzzy-RHONN controller* is shown in Fig. (1).

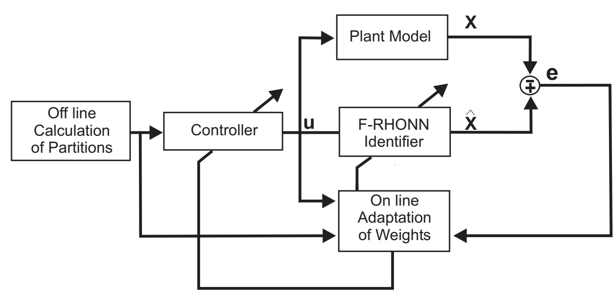

Fig. 1. Overall scheme of the proposed indirect adaptive neuro-fuzzy control system.

The following mild assumptions are also imposed on (26), to guarantee the existence and uniqueness of solution for any finite initial condition and $u \in U$.

Proposition 1: Given a class U of admissible inputs, then for any $u \in U$ and any finite initial condition, the state trajectories are uniformly bounded for any finite $T > 0$. Hence, $|x(T)| < \infty$.

Proposition 2: The vector fields f, g_i, $i = 1, 2, ..., n$ are continuous with respect to their arguments and satisfy a local Lipchitz condition so that the solution $x(t)$ of (26) is unique for any finite initial condition and $u \in U$.

We are using an affine in the control fuzzy dynamical system, which approximates the system in (26) and uses two fuzzy subsystem blocks for the description of $f(x)$ and $G(x)$ as follows

$$f(\chi) = A\chi + \sum \bar{f}^{l_1,\ldots,l_n}_{j_1,\ldots,j_n} \times I^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(\chi) \quad (27)$$

$$g_i(\chi) = \sum (\bar{g}_i)^{l_1,\ldots,l_n}_{j_1,\ldots,j_n} \times I_1{}^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(\chi) \quad (28)$$

where the summation is carried out over the number of all available fuzzy rules, I, I_1 are appropriate fuzzy rule indicator functions and the meaning of indices $\bullet^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}$ has already been described in Section II.

According to Lemma 1, every indicator function can be approximated with the help of a suitable HONNF. Therefore, every I, I_1 can be replaced with a corresponding HONNF as follows

$$f(\chi) = A\chi + \sum \bar{f}^{l_1,\ldots,l_n}_{j_1,\ldots,j_n} \times N^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(\chi) \quad (29)$$

$$\bar{g}_i(\chi) = \sum (\bar{g}_i)^{l_1,\ldots,l_n}_{j_1,\ldots,j_n} \times N_1{}^{l_1,\ldots,l_n}_{j_1,\ldots,j_n}(\chi) \quad (30)$$

where N, N_1 are appropriate HONNFs.

In order to simplify the model structure, since some rules result to the same output partition, we could replace the NNs associated to the rules having the same output with one NN and therefore the summations in (29),(30) are carried out over the number of the corresponding output partitions. Therefore, the affine in the control fuzzy dynamical system in (27), (28) is replaced by the following equivalent affine Recurrent High Order Neural Network (RHONN), which depends on the centers of the fuzzy output partitions $\bar{f}_l$ and $\bar{g}_{i,l}$

$$\dot{\hat{\chi}} = A\hat{\chi} + \sum_{l=1}^{Npf} \bar{f} \times N_l(\chi) + \sum_{i=1}^{n} \left(\sum_{l=1}^{Npg_i} (\bar{g}_i)_l \times N_{1l}(\chi) \right) u_i \quad (31)$$

Or in a more compact form

$$\dot{\hat{\chi}} = A\hat{\chi} + XWS(\chi) + X_1 W_1 S_1(\chi) u \quad (32)$$

Where A is a $n \times n$ stable matrix which for simplicity can be taken to be diagonal as $A = diag[a_1, a_2, ..., a_n]$, X, X_1 are matrices containing the centers of the partitions of every fuzzy output variable of $f(x)$ and $g(x)$ respectively, $S(\chi), S_1(\chi)$ are matrices containing high order combinations of sigmoid functions of the state χ and W, W_1 are matrices containing respective neural weights according to (31) and (23). The dimensions and the contents of all the above matrices are chosen so that $XWS(\chi)$ is a $n \times 1$ vector and $X_1 W_1 S_1(\chi)$ is a $n \times n$ matrix. Without compromising the generality of the model we assume that the vector fields in (28) are such that the matrix G is diagonal. For notational simplicity we assume that all output fuzzy variables are partitioned to the same number, m, of partitions. Under these specifications X is a $n \times n \cdot m$ block diagonal matrix of the form $X = diag(X^1, X^2, \ldots, X^n)$ with each X^i being an m-dimensional raw vector of the form

$$X^i = [\,\bar{f}^i_1 \quad \bar{f}^i_2 \quad \cdots \quad \bar{f}^i_m\,]$$

where $\bar{f}^i_p$ denotes the center of the p-th partition of f_i. Also, $S(\chi) = [\,s_1(\chi) \quad \ldots \quad s_k(\chi)\,]^T$, where each $s_i(\chi)$ with $i = \{1, 2, ..., k\}$, is a high order combination of sigmoid functions of the state variables and W is a $n \cdot m \times k$ matrix with neural weights. W assumes the form $W = [\,W^1 \quad \cdots \quad W^n\,]^T$, where each W^i is a matrix $\left[w^i_{j\,l}\right]_{m \times k}$. X_1 is a $n \times n \cdot m$ block diagonal matrix $X_1 = diag({}^1X^1, {}^1X^2, \ldots, {}^1X^n)$ with each ${}^1X^i$ being an m-dimensional raw vector of the form

$${}^1X^i = [\,\bar{g}^{i,i}_1 \quad \bar{g}^{i,i}_2 \quad \cdots \quad \bar{g}^{i,i}_m\,],$$

where $\bar{g}^{i\,i}_k$ denotes the center of the k-th partition of g_{ii}. W_1 is a $m \cdot n \times n$ block diagonal matrix $W_1 = diag({}^1W^1, {}^1W^2, \ldots, {}^1W^n)$, where each ${}^1W^i$ is a column vector $\left[{}^1w^i_{j\,l}\right]_{m \times 1}$ of neural weights. Finally, $S_1(\chi)$ is a $n \times n$ diagonal matrix with each diagonal element $s_i(\chi)$ being a high order combination of sigmoid functions of the state variables.

According to the above definitions the configuration of the F-HONNF approximator is shown in Fig. (2). When the inputs are given into the fuzzy-neural network shown in Fig. (2), the output of layer IV gives indicator function outputs which activate the corresponding rules and are calculated by Eq. (23). At layer V, each node performs a fuzzy rule while layer VI gives the function output.

The approximator of indicator functions, has four layers. At layer I, the input nodes represent input and/re state measurable variables. At layer II, the nodes represent the values of the sigmoidal functions. At layer III, the nodes are the values of high order sigmoidal combinations. The links between layer III and layer IV are fully connected by the weighting factors $W = [\,W^1 \quad \cdots \quad W^n\,]^T$, the adjusted parameters. Finally, at layer IV the output represents the values of indicator functions.

It has to be mentioned here that the proposed neuro-fuzzy representation, finally given by (32), offers some advantages over other fuzzy or neural adaptive representations. Considering the proposed approach from the adaptive fuzzy system (AFS) point of view, the main advantage is that the proposed approach is much less vulnerable in initial AFS design assumptions because there is no need for a-priori information related to the IF part of the rules (type and centers of membership functions, number of rules). This information is replaced by the existence of HONNFs. Considering the proposed approach from the NN point of view, the final representation of the dynamic equations is actually a combination of High Order Neural Networks, each one being specialized in approximating a function related to a corresponding center of output state membership function. This way, instead of having one large HONNF trying to approximate "everything" we have many, probably smaller, specialized HONNFs. Conceptually, this strategy is expected to present better approximation results; this is also verified in the simulations section. Moreover, as it will be seen in section IV-C, due to the particular bond of each HONNF with one center of an output state membership function, the existence of the control law is assured by introducing a novel technique of parameter "hopping" in the corresponding weight updating laws.

B. Parametric uncertainty

We assume the existence of only parameter uncertainty, so, we can take into account that the actual system (26) can be modeled by the following neural form

$$\dot{\chi} = A\chi + XW^*S(\chi) + X_1 W_1^* S_1(\chi) u \quad (33)$$

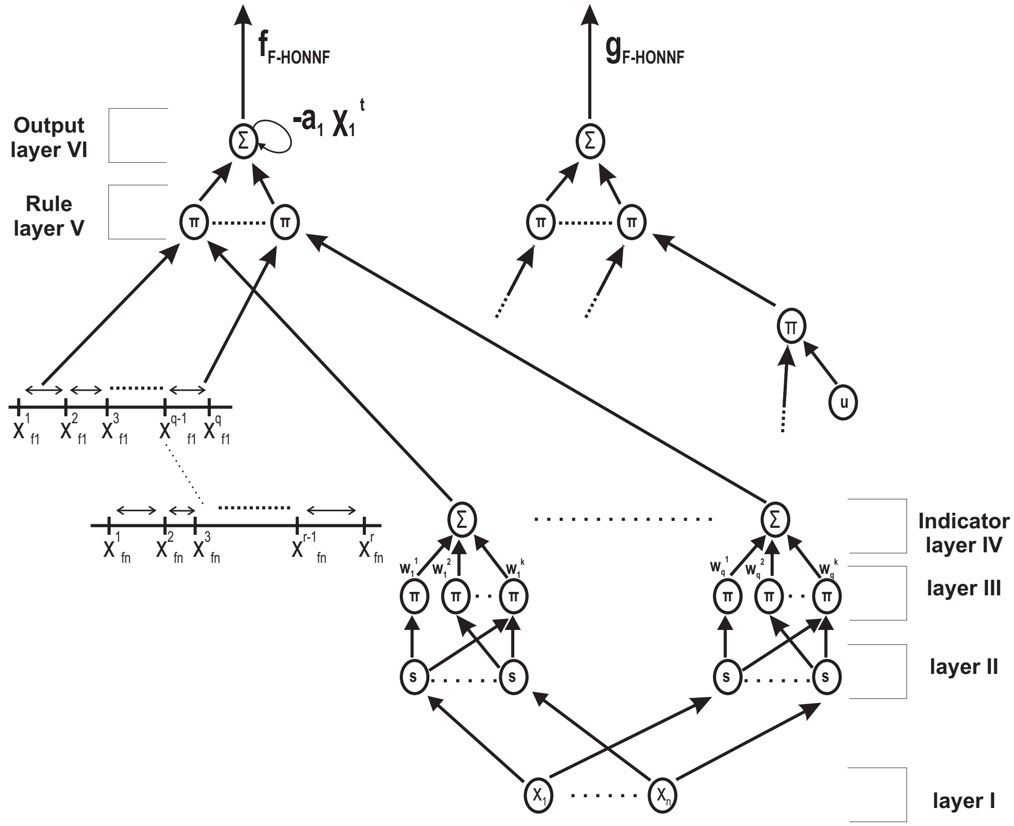

Fig. 2. Overall scheme of the proposed indirect adaptive neuro-fuzzy control system.

Define now, the error between the identifier states and the real states as

$$e = \hat{\chi} - \chi \tag{34}$$

Then from (32) and (34) we obtain the error equation

$$\dot{e} = Ae + X\tilde{W}S(\chi) + X_1\tilde{W}_1 S_1(\chi)u \tag{35}$$

Where $\tilde{W} = W - W^*$ and $\tilde{W}_1 = W_1 - W_1^*$.

Our objective is to find suitable control and learning laws to drive both e and χ to zero, while all other signals in the closed loop remain bounded. Taking u to be equal to

$$u = -\left[X_1 W_1 S_1(\chi)\right]^{-1} XWS(\chi) \tag{36}$$

and substituting it into (32) we finally obtain

$$\dot{\hat{\chi}} = A\hat{\chi} \tag{37}$$

In the next theorem the weight updating laws are given, which can serve both the identification and the control objectives provided that the updating of the weights of matrix W_1 is performed so that the existence of $[X_1 W_1 S_1(\chi)]^{-1}$ is assured.

Theorem 2: Consider the identification scheme given by 35. Provided that $[X_1 W_1 S_1(\chi)]^{-1}$ exists the learning law

a) For the elements of W^i

$$\dot{w}^i_{j\,l} = -\bar{f}^i_j p_i e_i s_l(\chi) \tag{38}$$

b) For the elements of ${}^1W^i$

$${}^1\dot{w}^i_{j\,1} = -\bar{g}^{i\,i}_j p_i e_i u_i s_i(\chi) \tag{39}$$

or equivalently ${}^1\dot{W}^i = -({}^1X^i)^T p_i e_i u_i s_i(\chi)$ with $i = 1, \ldots, n$, $j = 1, \ldots, m$, $l = 1, \ldots, k$ guarantees the following properties.

- $e, \hat{\chi}, \tilde{W}, \tilde{W}_1 \in L_\infty, \quad e, \hat{\chi} \in L_2$
- $\lim_{t\to\infty} e(t) = 0, \quad \lim_{t\to\infty} \hat{\chi}(t) = 0$
- $\lim_{t\to\infty} \dot{\tilde{W}}(t) = 0, \quad \lim_{t\to\infty} \dot{\tilde{W}}_1(t) = 0$

Proof: Consider the Lyapunov function candidate

$$V(e, \hat{\chi}, \tilde{W}, \tilde{W}_1) = \tfrac{1}{2}e^T Pe + \tfrac{1}{2}\hat{\chi}^T P\hat{\chi} + \tfrac{1}{2}tr\{\tilde{W}^T\tilde{W}\} + \tfrac{1}{2}tr\{\tilde{W}_1^T\tilde{W}_1\}$$

Where $P > 0$ is chosen to satisfy the Lyapunov equation

$$PA + A^T P = -I$$

Taking the derivative of the Lyapunov function candidate and taking into account (37) we get

$$\dot{V} = \tfrac{1}{2}e^T\left(A^T P + PA\right)e + \tfrac{1}{2}\hat{\chi}^T\left(A^T P + PA\right)\hat{\chi} + \left(\tfrac{1}{2}e^T PX\tilde{W}S + \tfrac{1}{2}e^T PX\tilde{W}S\right) + \left(\tfrac{1}{2}e^T PX_1\tilde{W}_1 S_1 u + \tfrac{1}{2}e^T PX_1\tilde{W}_1 S_1 u\right) + tr\{\dot{\tilde{W}}^T\tilde{W}\} + tr\{\dot{\tilde{W}}_1^T\tilde{W}_1\} \Rightarrow$$

$$\Rightarrow \dot{V} = -\tfrac{1}{2}e^T e - \tfrac{1}{2}\hat{\chi}^T\hat{\chi} + e^T PX\tilde{W}S + e^T PX_1\tilde{W}_1 S_1 u + \\ + tr\{\dot{\tilde{W}}^T \tilde{W}\} + tr\{\dot{\tilde{W}}_1^T \tilde{W}_1\} \Rightarrow$$

$$\Rightarrow \dot{V} = -\tfrac{1}{2}e^T e - \tfrac{1}{2}\hat{\chi}^T\hat{\chi} \leq 0$$

when

$$tr\{\dot{\tilde{W}}^T \tilde{W}\} = -e^T PX\tilde{W}S$$

$$tr\{\dot{\tilde{W}}_1^T \tilde{W}_1\} = -e^T PX_1\tilde{W}_1 S_1 u$$

Then, taking into account the form of W and W_1 the above equations result in the element wise learning laws given in (38),(39). These laws can also be written in the following compact form

$$\dot{W} = -X^T PeS^T \tag{40}$$

$$\dot{W}_1 = -X_1^T PEUS_1^T \tag{41}$$

Where E and U are diagonal matrices such that $E = diag(e_1, \ldots e_n)$ and $U = diag(u_1, \ldots u_n)$.

Using the above Lyapunov function candidate V and proving that $\dot{V} \leq 0$ all properties of the theorem are assured [31]. ■

Remark 1: The control law (36) can be also extended to the following form

$$u = -\left[X_1 W_1 S_1(\chi)\right]^{-1}\left[XWS(\chi) + kx\right] \tag{42}$$

where k is appropriate positive definite diagonal gain matrix. It can be easily verified that with this control law the negativeness of the derivative of the Lyapunov function is further enhanced. Therefore, term kx is actually acting as a robustifying term.

Proof: Indeed, by using the extended control law (42) the state estimate dynamics become

$$\dot{\hat{\chi}} = A\hat{\chi} - kx. \tag{43}$$

Then, using the weight updating laws given in theorem 2 the derivative of the Lyapunov function becomes

$$\dot{V} = -\tfrac{1}{2}e^T e - \tfrac{1}{2}\hat{\chi}^T\hat{\chi} - xKP\hat{\chi}$$
$$\Rightarrow \dot{V} = -\tfrac{1}{2}\|e\|^2 - \tfrac{1}{2}\|\hat{x}\|^2 - \hat{x}^T KP\hat{\chi} + e^T KP\hat{x}$$
$$\Rightarrow \dot{V} \leq -\tfrac{1}{2}\|e\|^2 - \tfrac{1}{2}\|\hat{x}\|^2 - \lambda_{min}(KP)\|\hat{x}\|^2 + \|e\|\,\|KP\|\,\|\hat{\chi}\|$$
$$= -\left[\,\|e\| \quad \|\hat{\chi}\|\,\right]\begin{bmatrix} 1/2 & -\|KP\| \\ 0 & 1/2 + \lambda_{min}(KP)\end{bmatrix}\begin{bmatrix}\|e\| \\ \|\hat{\chi}\|\end{bmatrix} < 0$$

■

C. Introduction to the parameter hopping

The weight updating laws presented previously in Section IV-B are valid when the control law signal in (36) exists. Therefore, the existence of $\left[X_1 W_1 S_1(\chi)\right]^{-1}$ has to be assured. Since $S_1(\chi)$ is diagonal with its elements $s_i(\chi) \neq 0$ and $X_1,\ W_1$ are block diagonal the existence of the inverse is assured when ${}^1X^i \cdot {}^1W^i \neq 0$, $\forall i = 1, \ldots n$. Therefore, W_1 has to be confined such that $\left|{}^1X^i \cdot {}^1W^i\right| \geq \theta_i > 0$, with θ_i being a design parameter. In case the boundary defined by the above confinement is nonlinear the updating W_1 can be modified by using a projection algorithm [31]. However, in our case the boundary surface is linear and the direction of updating is normal to it because $\nabla\left[{}^1X^i \cdot {}^1W^i\right] = {}^1X^i$. Therefore, the projection of the updating vector on the boundary surface is of no use. Instead, using concepts from multidimensional vector geometry we modify the updating law such that, when the weight vector approaches (within a safe distance θ_i) the forbidden hyper-plane ${}^1X^i \cdot {}^1W^i = 0$ and the direction of updating is toward the forbidden hyper-plane, it introduces a *hopping* which drives the weights in the direction of the updating but on the other side of the space, where here the weight space is divided into two sides by the forbidden hyper-plane. This procedure is depicted in Fig. 3, where a simplified 2-dimensional representation is given. Theorem 3 below introduces this *hopping* in the updating law.

Theorem 3: Consider the control scheme (35), (36), (37). The updating law:

a) *For the elements of W^i given by (38)*

b) *For the elements of ${}^1W^i$ given by the modified form:*

$${}^1\dot{W}^i = -({}^1X^i)^T p_i e_i u_i s_i(\chi) \quad if\ \left|{}^1X^i \cdot {}^1W^i\right| > \theta_i > 0$$
$$or \quad \left|{}^1X^i \cdot {}^1W^i\right| = \theta_i\ and \quad {}^1X^i \cdot {}^1\dot{W}^i \leq 0$$
$${}^1\dot{W}^i = -({}^1X^i)^T p_i e_i u_i s_i(\chi) - \\ -\frac{2}{tr\{({}^1X^i)^T\,{}^1X^i\}}\,{}^1X^i\,{}^1W^i\,({}^1X^i)^T \quad otherwise$$

guarantees the properties of theorem 2 and assures the existence of the control signal.

Proof: In order the properties of theorem 2 to be valid it suffices to show that by using the modified updating law for ${}^1W^i$ the negativeness of the Lyapunov function is not compromised. Indeed the ***if*** part of the modified form of ${}^1\dot{W}^i$ is exactly the same with (39) and therefore according to theorem 2 the negativeness of V is in effect. The *if* part is used when the weights are at a certain distance (condition if $\left|{}^1X^i \cdot {}^1W^i\right| > \theta_i$)from the forbidden plane or at the safe limit (condition $\left|{}^1X^i \cdot {}^1W^i\right| = \theta_i$) but with the direction of updating moving the weights far from the forbidden plane (condition ${}^1X^i \cdot {}^1\dot{W}^i \leq 0$).

In the ***otherwise*** part of ${}^1\dot{W}^i$, term $-\frac{2}{tr\{({}^1X^i)^T\,{}^1X^i\}}\,{}^1X^i\,{}^1W^i\,({}^1X^i)^T$ determines the magnitude of weight *hopping*, which as explained later and is depicted in Fig. 4 has to be two times the distance of the current weight vector to the forbidden hyper-plane. Therefore the ***existence*** of the control signal is assured because the weights never reach the forbidden plane. Regarding the ***negativeness*** of $\dot{V}$ we proceed as follows.

Let that ${}^1W^{*i}$ contains the initial values of ${}^1W^i$ provided from the identification part such that $\left|{}^1X^i \cdot {}^1W^{*i}\right| >> \theta_i$ and that ${}^1\tilde{W}^i = {}^1W^i - {}^1W^{*i}$. Then, the weight hopping can be equivalently written with respect to ${}^1\tilde{W}^i$ as $-2\theta_i{}^1\tilde{W}^i/\|{}^1\tilde{W}^i\|$. Under this consideration the modified updating law is rewritten as ${}^1\dot{W}^i = -({}^1X^i)^T p_i e_i u_i s_i(\chi) - 2\theta_i{}^1\tilde{W}^i/\|{}^1\tilde{W}^i\|$. With this updating law it can be easily verified that $\dot{V} = -\frac{1}{2}e^T e - \frac{1}{2}\hat{\chi}^T\hat{\chi} - \Theta$, with Θ being a positive constant expressed as $\Theta = \sum 2\theta_i \left(({}^1\tilde{W}^i)^T)^1\tilde{W}^i) \right) / \|{}^1\tilde{W}^i\|$, where the summation includes all weight vectors which require hopping. Therefore, the negativeness of $\dot{V}$ is actually enhanced. ■

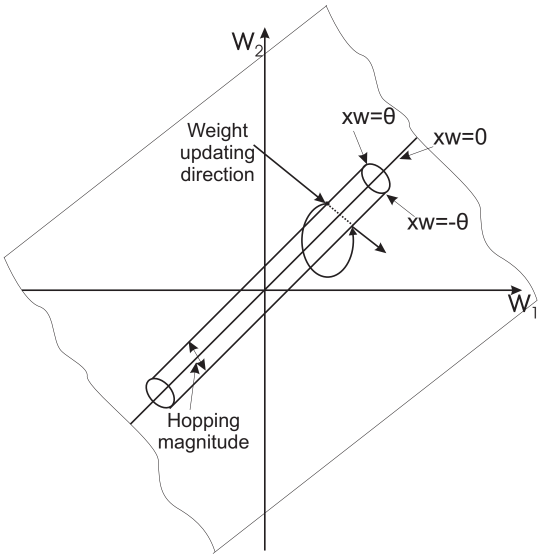

Fig. 3. Pictorial Representation of parameter hopping)

1) Vectorial proof of parameter hopping: In selecting the terms involved in parameter hopping we start from the vector definition of a line, of a plane and the distance of a point to a plane. The equation of a line in vector form is given by $r = a + \lambda t$, where a is the position vector of a given point of the line, t is a vector in the direction of the line and λ is a real scalar. By giving different numbers to λ we get different points of the line each one represented by the corresponding position vector r. The vector equation of a plane can be defined by using one point of the plane and a vector normal to it. In this case $r \cdot n = a \cdot n = d$ is the equation of the plane, where a is the position vector of a given point on the plane, n is a vector normal to the plane and d is a scalar. When the plane passes through zero, then apparently $d = 0$. To determine the distance of a point B with position vector b from a given plane we consider Fig. 4 and combine the above definitions as follows. Line BN is perpendicular to the plane and is described by vector equation $r = b + \lambda n$, where n is the normal to the plane vector. However, point N also lies on the plane and in case the plane passes through zero

$$r \cdot n = 0 \Rightarrow (b + \lambda n) \cdot n = 0 \Rightarrow \lambda = \frac{-b \cdot n}{\|n\|}.$$

Apparently, if one wants to get the position vector of B′ (the symmetrical of B in respect to the plane), this is given by

$$r = b - 2\frac{b \cdot n}{\|n\|}n.$$

In our problem $b = {}^1W^i$, our plane is described by the equation ${}^1X^i \cdot {}^1W^i = 0$ and as it has already been mentioned the normal to it is the vector ${}^1X^i$.

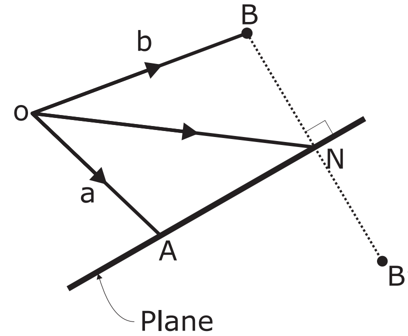

Fig. 4. Vector explanation of parameter hopping)

V. Simulation and Experimental Results

To demonstrate the potency of the proposed scheme we present simulation and experimental results. First, the function approximation abilities of the proposed technique are compared with those of a well established approach of adaptive fuzzy system definition for function approximation (see Eq. (2)). The simulations are carried out on the approximation of a nonlinear function appearing in the inverted pendulum benchmark problem. Next, the benchmark problem of Van der Pol oscillator is considered and two simulation results are presented. The first shows off the minimal parameter requirements of the proposed method when applied on this benchmark example, which would require a very large number of rules to be tackled by conventional fuzzy logic approach. The second presents the regulation of the Van der Pol system by using the proposed approach in comparison to a well established fuzzy adaptive control approach [2]. Due to the Brunovsky canonical form of the system the proposed method operates in a reduced model order, which although performing fairly well, does not permit to show off its full potential. The full potential of the method is demonstrated in the next simulation, where the proposed method is compared with the well known RHONN approach [32] in approximating and regulating a Dc Motor described by nonlinear equations. In this case both modeling approaches assume a generic affine in the control form. Finally, the proposed method was implemented and used to regulate a real DC Motor in an experimental set-up.

A. Comparison of function approximation abilities

Let the well known problem of the control of an inverted pendulum. Its dynamical equations can assume the following Brunovsky canonical form [48]

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x) + g(x)u \end{aligned} \tag{44}$$

where $x_1 = \theta$ and $x_2 = \dot{\theta}$ are the angle from the vertical position and the angular velocity respectively. $f(x)$ assumes the following form

$$f(x_1, x_2) = \frac{g \sin x_1 - \frac{mlx_2^2 \cos x_1 \sin x_1}{m_C + m}}{l\left(\frac{4}{3} - \frac{m\cos^2 x_1}{m_C + m}\right)} \tag{45}$$

where $g = 9.8\ m/s^2$ is the acceleration due to gravity, m_c is the mass of the cart, *m* is the mass of the pole, and *l* is the half-length of the pole. We choose $m_c = 1\ kg$, $m = 0.1\ kg$, and $l = 0.5\ m$ in the following simulation. In this case we also have that $|x_1| \leq \pi/6$ and $|x_2| \leq \pi/6$.

It is our intention to compare the approximation abilities of the proposed Neuro-Fuzzy approach with Wang [2] adaptive Fuzzy approach. To this end we assume that $f(x)$ can be approximated using Wang's approach and Eq. (2) or alternatively by the *XWS* term of Eq. (32) in the proposed approach. The weight updating laws are chosen to be: For the Wang approach ([2], page 115)

$$\dot{\theta}_f = -\gamma_1 e^T P b_c \xi(x) \tag{46}$$

where only the simplified approach, without parameter projection case was necessary to be used.

For the proposed F-HONNF approach we use the following adaptive law:

$$\dot{W} = -X^T P e S^T \tag{47}$$

The experimental data were obtained as follows: Based on Wang's input variables limits and fuzzy partition we created an artificial stair-like signal shown in Fig. (6). Input variables x_1 and x_2 assume values in the interval $[-\pi/6, \pi/6]$.

Taking 5 samples from x_1 and 100 samples from x_2 we obtain 500 samples of $f(x_1, x_2)$ presenting the stair discontinuities when x_1 changes values. For the construction of ξ_i functions used in Eq. (2) and given in Eq. (3) we used the fuzzy membership partitions and the final rules characterizing $f(x_1, x_2)$ and shown in Fig. (5), which comprises 25 fuzzy rules carefully chosen and given by Wang in [2] (page 129). Under these design specifications Eq. (2) assumes 25 adjustable weights.

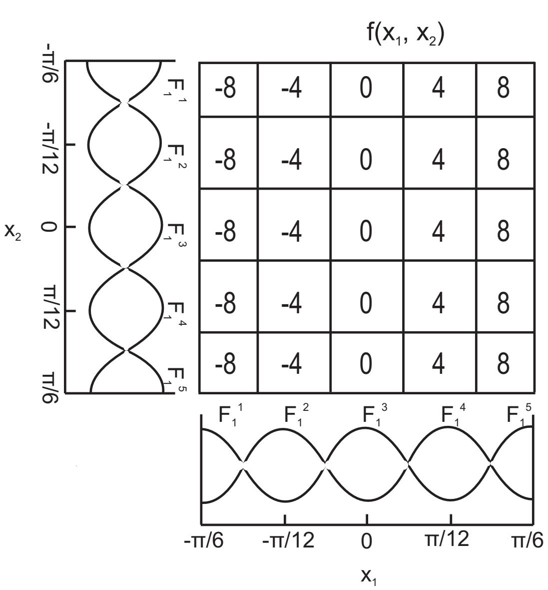

Fig. 5. Linguistic fuzzy rules for $f(x_1, x_2)$

In order our model to be equivalent with regard to adjustable parameters we have chosen 5 centers for the fuzzy output variables partition (-8, -4, 0, 4, 8) and 5 high order sigmoidal terms $(s(x_1), s(x_2), s(x_1) \cdot s(x_2), s^2(x_1), s^2(x_2))$ in each HONNF. This configuration also assumes 25 adjustable weights. Terms $\gamma_1 P b_c$ in Eq. (46) and P (the updating learning rates) in Eq. (47) were chosen to have the same values. Fig. (6) shows the approximation abilities of (2) with the updating law of (46) while Fig. (7) shows the performance of the proposed approach with the updating law of (47). The mean squared error (MSE) for Wang's approach was measured to be $6.24 \cdot 10^{-4}$, while for the proposed approach was $1.25 \cdot 10^{-5}$, demonstrating a significant (order of magnitude) increase in the approximation performance, although in our approach no a-priori information regarding the inputs were used.

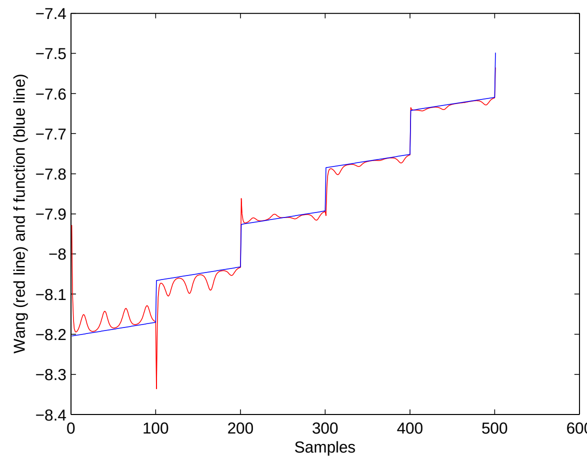

Fig. 6. Approximation of the *f* function with Eq.(2)

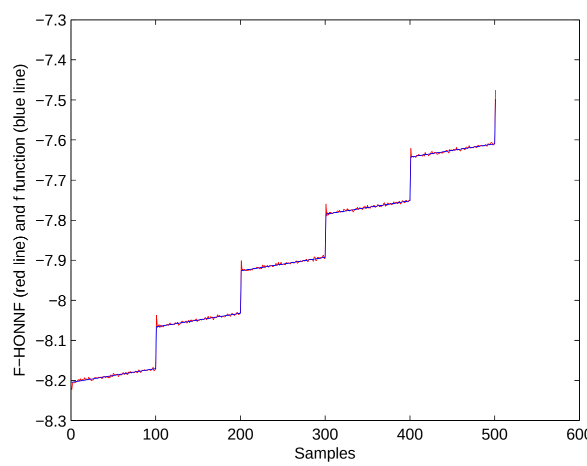

Fig. 7. Approximation of the *f* function with the proposed approach.

B. Demonstrating the minimum parameter requirements

Van der Pol oscillator is usually used as a simple benchmark problem for testing identification and control schemes. It's dynamical equations are of the same Brunovsky form as (44) and are given by

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_2 \cdot \left(a - x_1^2\right) \cdot b - x_1 + u \end{aligned} \tag{48}$$

The approximation of the dynamical equations using conventional fuzzy system approach requires a very large number of fuzzy rules. Even after a careful selection of rules, this number could be chosen to be almost 2500 for very accurate fuzzy representation [47]. This in turn would lead to a parameter explosion when using an adaptive scheme like that of Eq. (2).

We are using the proposed approach with Eq. (32) to approximate Van der Pol dynamics. The proposed Neuro-Fuzzy model was chosen to use 5 output partitions of f and 2 output partitions of g. The number of high order terms used in HONNF's were chosen to be 3. Therefore, the number of adjustable weights is 30 for f and 6 for G, because in this case only the second equation is assumed to be influenced by the input. Therefore, the total number, 36, of the adjustable weights is much smaller than the required one in the conventional adaptive fuzzy approach. Consequently, the required training time of the proposed approach is much smaller. Numerical training data were obtained by using Eq. (48) with initial conditions, $x_1(0) = x_2(0) = 1$, and a persistently exciting input $u = 1 + 0.8\sin(0.001t)$. Fig. (8) and Fig. (9) shows the approximating errors of the proposed scheme for x_1 and x_2 respectively, which approaches zero very fast. Also, Fig. (10) and Fig. (11) shows the approximation for the variables x_1 and x_2 respectively.

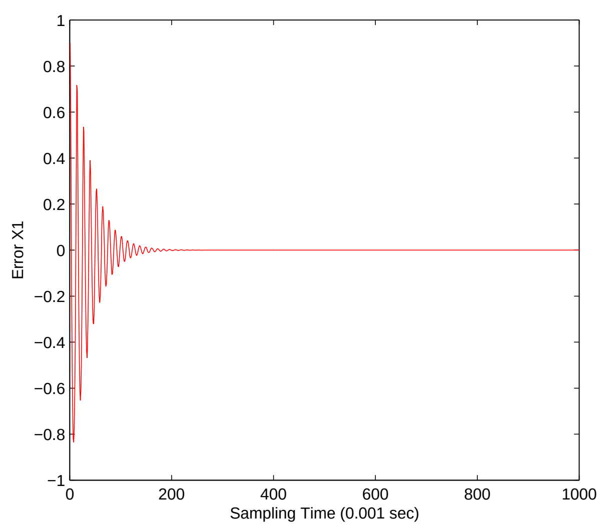

Fig. 8. Approximation Error for variable x_1

C. Regulation of the Van der Pol oscillator

The regulation of the Van der Pol oscillator (48) is considered next. It has to be mentioned here that both inverted pendulum and Van der Pol dynamic equations, which are usually used as benchmark problems, assume a Brunovsky canonical form with the control input appearing in the last equation. Our modeling approach does not bear to this restriction and assumes a more generic affine in the control form, which however requires inputs in all dynamic equations. Therefore,

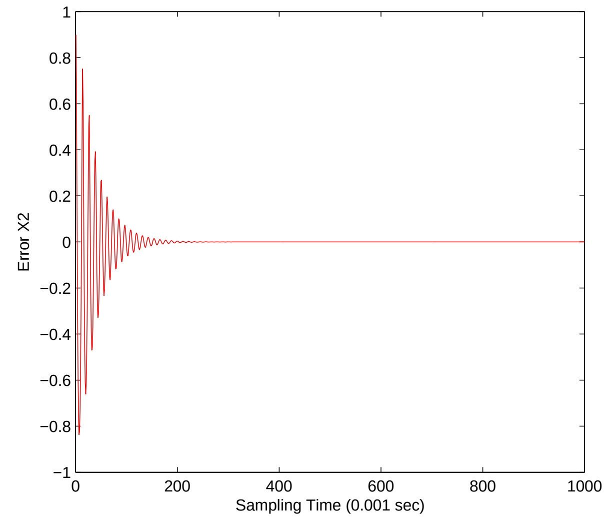

Fig. 9. Approximation Error for variable x_2

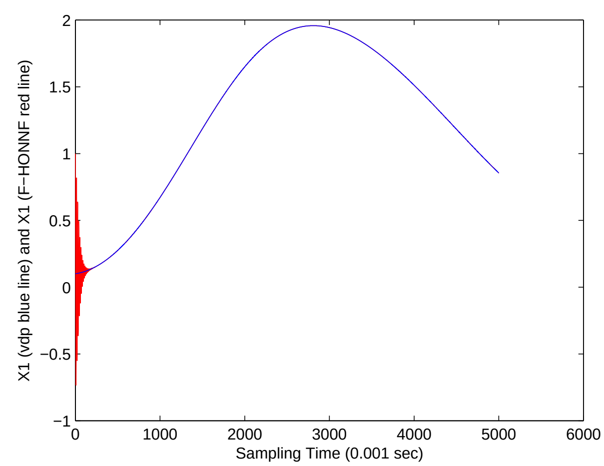

Fig. 10. Approximation for variable x_1

a comparison can be made only if our technique assumes a reduced model order form; in the problem under study the F-HONNF is reduced to a first order model (1 state).

The proposed technique is compared to the indirect fuzzy adaptive control appearing in [2], which is frequently used for comparison reasons in the recent literature. In this approach a feedback linearizing certainty equivalent control law is used assuming the form ([2], chapter 8, page 108)

$$u = \frac{1}{\hat{g}(x_1, x_2)} \left[-\hat{f}(x_1, x_2) + \dot{x}_{2d} + k^T e \right] \tag{49}$$

where $\hat{f}, \hat{g}$ are fuzzy adaptive approximations of f and g respectively, according to Eq. (2). e is a vector containing the errors between their respective desired values and the states x_1, x_2, $k^T = [k_1, k_2]$ is a gain vector and $\dot{x}_{2d} = \ddot{x}_{1d}$ is the desired value of $\dot{x}_2$. In such approximate feedback linearizing laws, the term $k^T e$ acts also as a robustifying term keeping the approximately linearized dynamics stable until the approximates $\hat{f}$ and $\hat{g}$ become sufficiently accurate [52]. In a

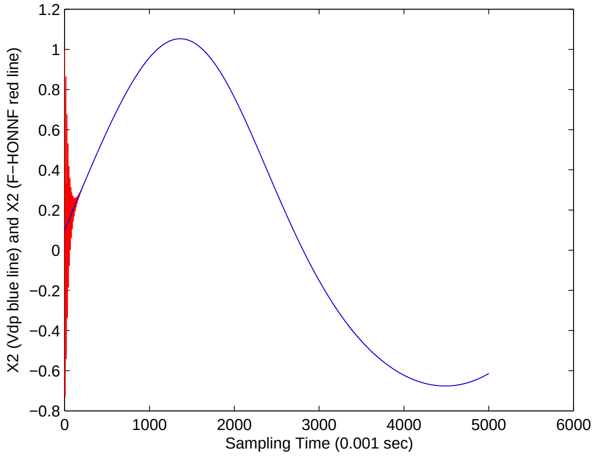

Fig. 11. Approximation for variable x_2

state regulation problem $x_{1d} = x_{2d} = 0$, so Eq. (49) is reduced to

$$u = \frac{1}{\hat{g}(x_1, x_2)} \left[-\hat{f}(x_1, x_2) - k_1 x_1 - k_2 x_2 \right]. \tag{50}$$

In simulating the above approach we assumed that $\hat{f}$ and $\hat{g}$ are estimated using Eq. (2), where x_1, x_2 are partitioned into 5 fuzzy membership values each. Therefore, for each of the functions 25 rules and consequently 25 weights in the corresponding weights vectors θ are required. Thus the total number of the adjustable weights amounts to 50. The weight updating is performed by using equations (8.42) - (8.46) of [2], where the design parameters are carefully chosen so that the regulation of the Van der pol benchmark is optimal. Thus, we choose $k_1 = 5$ and $k_2 = 4$ (so that $s^2 + k_1 s + k_2$ has roots in the open left half-plane) and $Q = diag(10, 10)$. Then, according to [2] we solve equation $\Lambda_c^T P + P\Lambda_c = -Q$ to obtain

$$P = \begin{bmatrix} 11.25 & 1.25 \\ 1.25 & 1.25 \end{bmatrix} \tag{51}$$

where $\Lambda_c = \begin{bmatrix} 0 & 1 \\ -k_2 & -k_1 \end{bmatrix}$

This P is positive definite. From the bounds of $f(x_1, x_2)$ and $g(x_1, x_2)$ we see that the range of $f(x_1, x_2)$ is larger than that of $g(x_1, x_2)$, therefore we choose the gains $g_1 = 5$ and $g_2 = 1$. Continuously, we choose the initial weights $\theta_f(0)$ randomly in the interval $[-1, 1]$ and $\theta_g(0)$ randomly in the interval $[0.8, 1.2]$. Finally, the tuning of the membership functions becomes such that the bounds of x_1 and x_2 are covered. Thus, the membership functions have the following form

$$\mu(x_i) = \exp\left[-\left(\frac{x_i + c_j}{d} \right)^2 \right] \tag{52}$$

where $c_j = [-1, -0.5, 0, 0.5, 1]$ (j=1,2,...,5) are the centers and $d = 0.15$ is the deviation.

The proposed method is also tested on the same problem. To this end, since the method assumes a general affine in the control form and not a Brunovsky one, it is assumed that the system is approximated by a first order nonlinear FHONNF model given by

$$\dot{\hat{\chi}}_2 = -\alpha \hat{\chi}_2 + XWS(x) + X_1 W_1 S_1(x) u \tag{53}$$

where α is a positive constant, X is a row vector of $m = 5$ centers of fuzzy partitions covering the range [-23, -11], $S(x)$ is a column vector with $k = 5$ high order sigmoidal terms and W is a $m \times k$ matrix with neural weights. Similarly, X_1 is a row vector of $m = 5$ centers of fuzzy partitions covering the range [0.8, 1.2] S_1 is a column vector with $k = 5$ high order sigmoidal terms and W_1 a matrix containing the corresponding neural weights. Finally, the parameters of the sigmoidals that have been used are $\alpha_1 = \alpha_2 = 0.1$, $\beta_1 = \beta_2 = 1$ and $\gamma_1 = \gamma_2 = 0$.

The control objective is to drive state x_2 to zero with x_1 being kept bounded. To this end, in order to have an equivalent comparison with the control law of (50) containing robustifying terms the control law (42) of remark 1 was used. In this example the control law is written as

$$u = -\left[X_1 W_1 S_1(\chi) \right]^{-1} \left[XWS(\chi) + kx_2 \right] \tag{54}$$

where $k = 5$ and the other quantities were defined in (53) above. Figure (12) shows the convergence of state x_2 to zero using the approach in [2] and the proposed approach respectively, while figure (13) shows the evolution of control inputs. It can be observed that although the proposed approach operates in a reduced model order form it performs much better than the conventional approach and presents much smoother evolution of the state x_2 to zero.

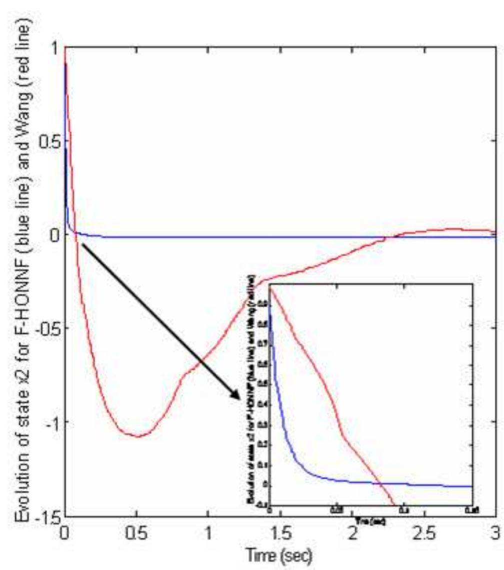

Fig. 12. Evolution of state x_2 for F-HONNF and Wang approach.

D. DC Motor Identification and Control

In this section we present simulations, where the proposed approach is applied to solve the problem of controlling the

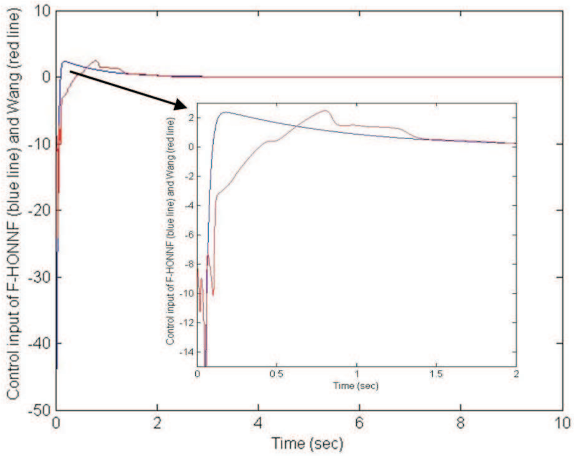

Fig. 13. Evolution of control input u for F-HONNF and Wang approach.

speed of a 1 KW DC motor with a normalized model described by the following dynamical equations [46]

$$\begin{aligned} T_a \frac{dI_a}{dt} &= -I_a - \Phi\Omega + V_a \\ T_m \frac{d\Omega}{dt} &= \Phi I_a - K_0\Omega - m_L \\ T_f \frac{d\Phi}{dt} &= -I_f + V_f \\ \Phi &= \frac{aI_f}{1+bI_f} \end{aligned} \quad (55)$$

The states are chosen to be the armature current, the angular speed and the stator flux $x = [\, I_a \quad \Omega \quad \Phi \,]^T$. As control inputs the armature and the field voltages $x = [\, V_a \quad V_f \,]^T$ are used. With this choice, we have

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -\frac{1}{T_a}x_1 - \frac{1}{T_a}x_2x_3 \\ \frac{1}{T_m}x_1x_3 - \frac{K_0}{T_m}x_2 - \frac{m_L}{T_m} \\ -\frac{1}{T_f}\frac{x_3}{a-\beta x_3} \end{bmatrix} + \begin{bmatrix} \frac{1}{T_a} & 0 \\ 0 & 0 \\ 0 & \frac{1}{T_f} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (56)$$

which is of a nonlinear, affine in the control form. The regulation problem of a DC motor is translated as follows: Find a state feedback to force the angular velocity and the armature current to go to zero, while the magnetic flux varies.

When Φ is considered constant, the above nonlinear 3rd order system can be linearized and reduced to a second order form having 2 states ($x_1 = I_a$ and $x_2 = \Omega$), with the value Φ being included as a constant parameter. Inspired by that, we first assume that the system is described, within a degree of accuracy, by a 2nd order ($n = 2$) nonlinear neuro-fuzzy system of the form (32), where $x_1 = I_a$ and $x_2 = \Omega$. Coefficients a_i in matrix A of (32) were chosen to be $a_i = 15$. The number of fuzzy partitions in X was chosen to be $m = 5$ and the range of f_1 [-182.5667 , 0], f_2 [-19.3627 , 30.0566]. The depth of high order terms was $k = 2$ (only first order sigmoidal terms $S(x_1)$, $S(x_2)$ were used). The number of fuzzy partitions of each $g_{i\,i}$ in X_1 is $m = 1$ and the range of g_{11} is [148 , 150] and of g_{22} is [42 , 44]. The parameters of the sigmoidals that have been used are $\alpha_1 = 0.4$, $\alpha_2 = 5$, $\beta_1 = \beta_2 = 1$ and $\gamma_1 = \gamma_2 = 0$. In the simulations carried out, the actual system is simulated by using the complete set of equations (56). The produced control law (36) is applied on this system, which in turn produces states x_1, x_2, which in the sequel are used for the computation of the estimation errors that are employed by the updating laws.

TABLE I
PARAMETER VALUES FOR THE DC MOTOR.

Parameter	Value
$1/T_a$	148.88
$1/T_m$	42.91
K_0/T_m	0.0129
T_f	31.88
T_L	0.0
a	2.6
β	1.6

We simulated a 1KW DC motor with parameter values that can be seen in Table I. Our two stage algorithm, was applied.

For comparison purposes we test the identification abilities of the proposed F-HONNF model against the conventional RHONN approximator presented in [32] using equivalent parameters regarding learning rate and number of high order terms used. Fig. 14 shows the performance of the proposed scheme (blue line) against the corresponding performance of RHONN (red line). In the embedded figure a detailed comparison between the two methods for the first iterations is presented, where the graph is adjusted to the scale of the lower error values (those of the F-HONNF model). The mean square error (MSE) was measured to be $5,87 * 10^{-5}$ for the proposed scheme and $1,18 * 10^{-2}$ for RHONN showing that the proposed scheme performs much better.

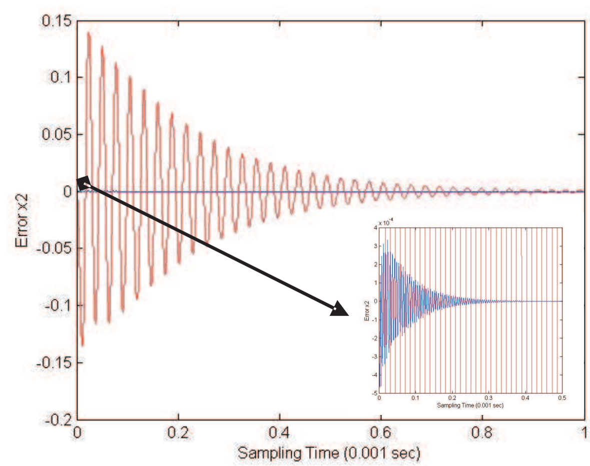

Fig. 14. Evolution of e_2.

In the control phase, we assumed that the system variables have the initial values $\Omega = 0.1$, $I_a = 0.1$, $\Phi = 0.98$. The proposed feedback control law and the corresponding control law of [32] were applied, with the corresponding initial weight values resulted from the identification phase. Figures 15, 16 give the evolution of the angular velocity and armature current respectively, for F-HONNF (blue line) and RHONN (red line). As can be seen, both Ω and I_a converge to zero very fast as desired and the corresponding mean squared errors are 0.0017 and 0.0135 for x_1 (F-HONNF Vs RHONN approach) and 0.0013 and 0.0094 for x_2 (F-HONNF Vs RHONN approach), demonstrating a significant improvement when the proposed method is used.

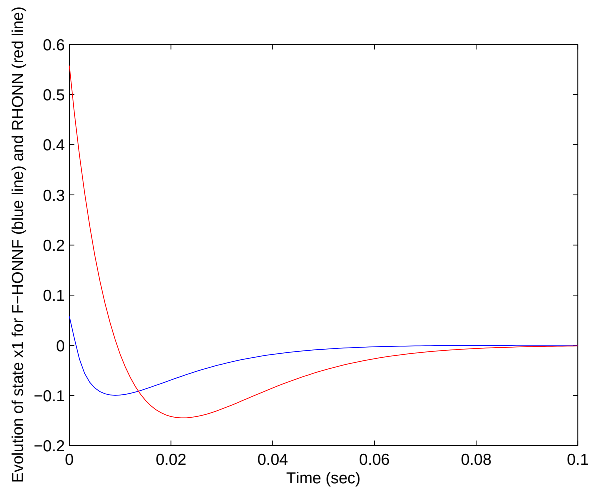

Fig. 15. Convergence of the angular velocity to zero from 0.1 initially.

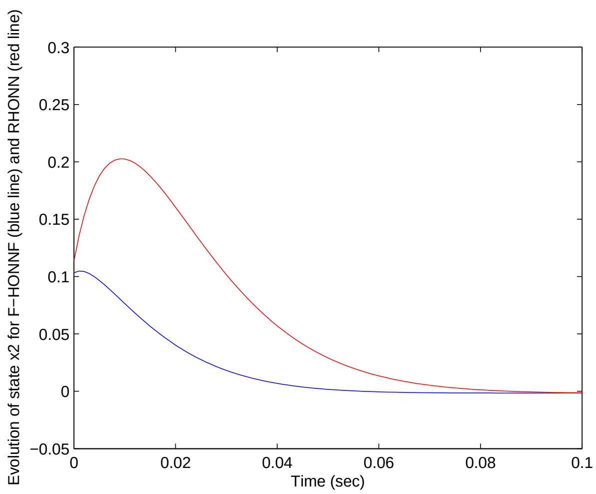

Fig. 16. Convergence of the armature current to zero from 0.1 initially.

E. Experimental setup and verification

The proposed method was also verified on an experimental setup. It consists of two separately excited dc machines (one of which is operating as a motor providing the second one with mechanical power, which in turn is operating as a generator supplying a resistor bank), a Digital Signal Processor (DSP) card, dc power supplies, dc-dc converters, a rotary encoder, shunt resistors, and a personal computer. It is represented schematically in Fig. 17. A data acquisition card (DAC) mounted in the PC is interfaced with the assembly module to acquire the speed and the current information. The data acquisition in the PC and data acquisition and control in the DSP card is accomplished using C++ programming. The setup can be used for armature and field weakening control of the separately excited DC motor but it is also used to test the proposed control algorithm, which assumes control of both armature and field excitation. Motor data are given in table II. The control algorithm performs weight updating and control input calculation according to equations (38),(39),(36). Similarly to the simulations given in the previous section we assume a second order affine in the control F-HONNF representation of the system. The number of fuzzy partitions

TABLE II
MOTOR DATA USED IN THE SETUP.

Parameters	Value
Armature resistance R	0.81 ohms
Armature inductance L	57 mH
Moment of inertia J	0.39 kg-m
Viscous friction coefficient	0.0164 Nm./rad/Sec
Back e.m.f. coefficient K	1.55 V/rad/sec

in X was chosen to be $m = 5$ and the depth of high order terms was $k = 5$ (up to second order sigmoidal terms $S(x_i)$, were used). The number of fuzzy partitions of each g_{ii} in X_1 is $m = 3$. The control objective was to drive both angular velocity and armature current to zero starting from 0.3 of their nominal values. The results of the experiment are shown in Figs. 18 and 19, verifying the effectiveness of the proposed technique.

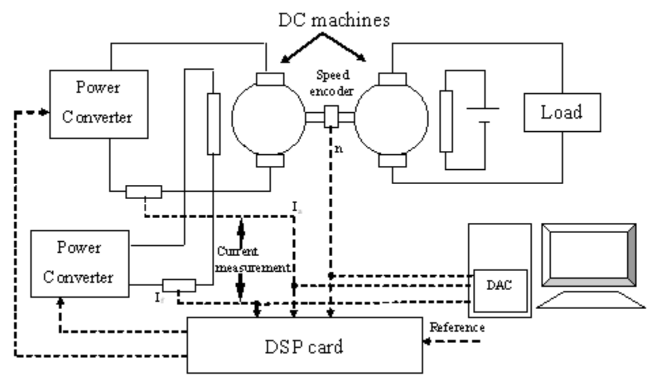

Fig. 17. Schematic diagram of the experimental setup

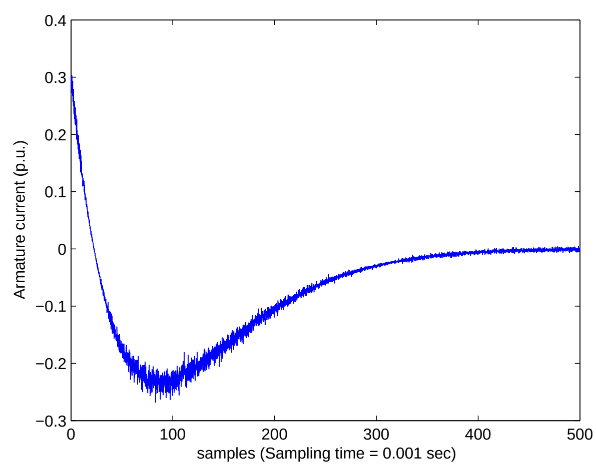

Fig. 18. Convergence of current to zero from 0.3 (p.u) initially

VI. CONCLUSION

An indirect adaptive control scheme was considered in this paper, aiming at the regulation of non linear unknown plants. The approach is based on a new Neuro-Fuzzy Dynamical Systems definition, which uses the concept of Adaptive Fuzzy Systems (AFS) operating in conjunction with High Order

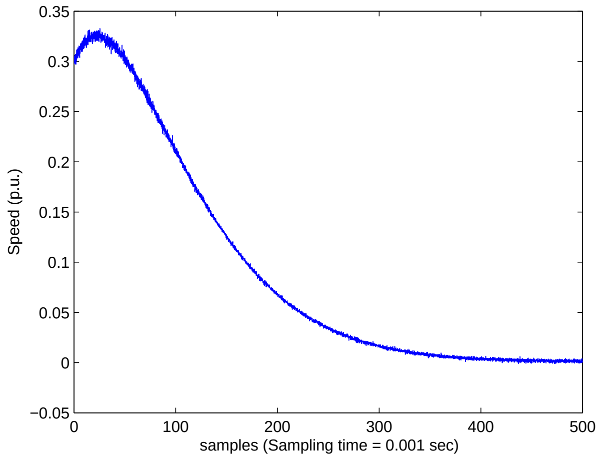

Fig. 19. Convergence of speed to zero from 0.3 (p.u) initially

Neural Network Functions (F-HONNFs). Since the plant is considered unknown, we first propose its approximation by a special form of an affine in the control fuzzy dynamical system (FDS) and in the sequel the fuzzy rules are approximated by appropriate HONNFs. Once the system is identified around an operation point is regulated to zero adaptively. The proposed scheme does not require a-priori experts' information on the number and type of input variable membership functions making it less vulnerable to initial design assumptions. Weight updating laws for the involved HONNFs are provided, which guarantee that both the identification error and the system states reach zero exponentially fast, while keeping all signals in the closed loop bounded. A method of parameter hopping assures the existence of the control signal and is incorporated in the weight updating law. Simulations illustrate the potency of the method by comparing its performance with this of established conventional approaches on benchmark problems. And finally, the applicability of the method was tested on a DC Motor system where it is shown that by following the proposed procedure one can obtain asymptotic regulation.

ACKNOWLEDGMENT

The authors wish to acknowledge the substantial help of Dr. A. Karlis from the laboratory of electrical machines in setting up the experiment for the control of the DC motor.

REFERENCES

[1] K. Hornic, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators", *Neural Networks*, vol. 2, pp. 359-366, 1989.

[2] L. Wang, *Adaptive Fuzzy Systems and Control*, Prentice Hall, Englewood Cliffs, NJ, 1994.

[3] K. M. Passino and S. Yurkovich, *Fuzzy Control*, Addison Wesley Longman, Menlo Park, CA, 1998.

[4] J. S. R. Jang, "ANFIS: Adaptive-network-based fuzzy inference system", *IEEE Trans. Syst. Man. Cyber.*, vol. 23, pp. 665-684, 1993.

[5] C.T. Lin, "A neural fuzzy control system with structure and parameter learning", *Fuzzy Sets and Systems*, vol. 70, pp. 183-212, 1995.

[6] K.B. Cho and B.H.Wang, "Radial basis function based adaptive fuzzy systems and their applications to system identification and prediction", *Fuzzy Sets and Systems*, vol. 83, pp. 325-339, 1996.

[7] C.F.Juang and C. T. Lin, "An on-line self-constructing neural fuzzy inference network and its applications", *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 12-32, 1998.

[8] R. P. Li and M. Mukaidono, "A new approach to rule learning based on fusion of fuzzy logic and neural networks", *IEICE Trans. Fuzzy Syst*, vol. E78-d, pp. 1509-1514, 1995.

[9] S. L. Chiu, "Fuzzy model identification based on cluster estimation", *Journal of Intelligent and Fuzzy Systems*, vol. 2, 1994.

[10] Y. H. Lin and G. A. Cunningham, "A new approach to fuzzy-neural system modelling", *IEEE Trans. Fuzzy Syst.*, vol. 3, pp. 190-197, 1995.

[11] C. F. Jang and C. T. Lin, "An online self-constructing neural fuzzy inference network and its applications", *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 12-32, 1998.

[12] S. Mitra and Y. Hayashi, "Neuro-fuzzy rule generation: survey in soft computing framework", *IEEE Trans. On Neural Networks*, vol. 11, pp. 748-768, 2000.

[13] B. S. Chen, C. H. Lee, and Y. C. Chang, "Tracking design of uncertain nonlinear siso systems: Adaptive fuzzy approach", *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 32-43, Feb. 1996.

[14] J. T. Spooner and K. M. Passino, "Stable adaptive control using fuzzy systems and neural networks", *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 339-359, June 1996.

[15] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks", *IEEE Trans. Neural Networks*, vol. 1, pp. 4-27, Feb.1990.

[16] M. M. Polycarpou and M. J. Mears, "Stable adaptive tracking of uncertain systems using nonlinearly parameterized online approximators", *Int. J. Control*, vol. 70, no. 3, pp. 363-384, 1998.

[17] G. A. Rovithakis and M. A. Christodoulou, "Adaptive control of unknown plants using dynamical neural networks", *IEEE Trans. Syst, Man, Cybern.*, vol. 24, , pp. 400-412 Mar. 1994.

[18] N. Golea, A. Golea, K. Benmahammed, "Stable indirect fuzzy adaptive control", *Fuzzy Sets and Systems*, vol. 137, pp. 353-366, 2003.

[19] T. Lina, C. Wang, H. Liub, "Observer-based indirect adaptive fuzzy-neural tracking control for nonlinear SISO systems using VSS and H_1 approaches", *Fuzzy Sets and Systems*, vol. 143, pp. 211-232, 2004.

[20] Y. Diao, K.M. Passino, Adaptive Neural/Fuzzy Control for Interpolated Nonlinear Systems, *IEEE Trans. on Fuzzy Systems*, Vol. 10, No. 5, pp. 583-595, Oct. 2002.

[21] R. Ordonez, K.M. Passino, "Adaptive Control for a Class of Nonlinear Systems with Time-Varying Structure", *IEEE Trans. on Automatic Control*, Vol. 46, No. 1, pp. 152-155, Jan.

[22] D. Sbarbaro, T.A. Johansen, "Analysis of Artificial Neural Networks for Pattern-Based Adaptive Control", *IEEE Trans. on Neural Networks*, vol. 17, no. 5, pp. 1184 - 1193, Sept. 2006.

[23] T. Hayakawa, W.M. Haddad, N. Hovakimyan, V. Chellaboina, "Neural network adaptive control for nonlinear nonnegative dynamical systems", *IEEE Trans. on Neural Networks*, vol. 16, no. 2, pp. 399 - 413, March 2005.

[24] S. G. Shuzhi, T. H. Thanh, "Semiglobal ISpS Disturbance Attenuation With Output Tracking via Direct Adaptive Design", *IEEE Trans. on Neural Networks*, vol. 18, no. 4, pp. 1129 - 1148, July 2007.

[25] N. Hovakimyan, E. Lavretsky, J. Y. Bong, A.J. Calise, "Coordinated decentralized adaptive output feedback control of interconnected systems", *IEEE Trans. on Neural Networks*, vol. 16, no. 1, pp. 185 - 194, Jan. 2005.

[26] S.S. Ge, W. Jing, "Robust adaptive neural control for a class of perturbed strict feedback nonlinear systems", *IEEE Trans. on Neural Networks*, vol. 13, no. 6, pp. 1409 - 1419, Nov. 2002.

[27] C. Lee and C. Teng, "Identification and Control of Dynamic Systems Using Recurrent Fuzzy Neural Networks", *IEEE Trans. Fuzzy Syst.*, vol. 8, pp. 349-366, 2000.

[28] M. Hojati and S. Gazor, "Hybrid Adaptive Fuzzy Identification and Control of Nonlinear Systems", *IEEE Trans. Fuzzy Syst.*, vol. 10, pp. 198-210, April 2002.

[29] H.N. Nounou, K.M. Passino , Stable Auto-Tuning of Hybrid Adaptive Fuzzy/Neural Controllers for Nonlinear Systems, Engineering Applications of Artificial Intelligence, Vol. 18, No. 3, pp. 317-334, April 2005.

[30] F. C. Chen and C. C. Liu, "Adaptively controlling nonlinear continuous-time systems using multilayer neural networks", *IEEE Trans. Automat. Contr.*, vol. 39, pp. 1306-1310, June 1994.

[31] P. Ioannou and B. Fidan, *Adaptive control tutorial*, SIAM: Advances in Design and Control Series, 2006.

[32] G.A. Rovithakis and M.A.Christodoulou, *Adaptive Control with Recurrent High Order Neural Networks (Theory and Industrial Applications)*, in Advances in Industrial Control, M.A.Grimble and M.A.Johnson, Springer Verlag London Limited, 2000.

[33] J. T. Spooner, M. Maggiore, R. Ordonez, and K. M. Passino, *Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximator Techniques*, John Wiley and Sons, NY, 2002.

[34] H. Nounou, K.M. Passino, Stable Auto-Tuning of Adaptive Fuzzy/Neural Controllers for Nonlinear Discrete-Time Systems, *IEEE Trans. on Fuzzy Systems*, Vol. 12, No. 1, pp. 70-83, Feb. 2004.

[35] J. H. Park, S.-H. Huh, S.-J. Seo and G.-T. Park, "Direct adaptive controller for nonaffine nonlinear systems using self-structuring neural networks", *IEEE Trans. on Neural Networks*, vol. 16, no. 2, pp. 4141-422, 2005.

[36] Y. Leu, W. Wang,and T. Lee, "Observer-Based Direct Adaptive Fuzzy-Neural Control for Nonaffine Nonlinear Systems", *IEEE Trans. on Neural Networks*, vol. 16, pp. 853-861, July 2005.

[37] J. Y. Bong, A.J. Calise, "Adaptive Control of a Class of Nonaffine Systems Using Neural Networks", *IEEE Trans. on Neural Networks*, vol. 18, no. 4, pp. 1149 - 1159, July 2007.

[38] G. L. Yih, Y. W. Wei, T. L. Tsu, "Observer-based direct adaptive fuzzy-neural control for nonaffine nonlinear systems", *IEEE Trans. on Neural Networks*, vol. 16, no. 4, pp. 853 - 861, July 2005.

[39] D. Sbarbaro, T.A. Johansen, "Analysis of Artificial Neural Networks for Pattern-Based Adaptive Control", *IEEE Trans. on Neural Networks*, vol. 17, no. 5, pp. 1184 - 1193, Sept. 2006.

[40] G. A. Rovithakis and M. A. Christodoulou, "Direct adaptive regulation of unknown nonlinear dynamical systems via dynamic neural networks", *IEEE Trans. Syst, Man, Cybern.*, vol. 25, pp. 1578-1594, Dec 1995.

[41] E. B. Kosmatopoulos and M. A. Christodoulou, "Recurrent neural networks for approximation of fuzzy dynamical systems", *Int. Journal of Intelligent Control and Systems*, vol. 1, pp. 223-233, 1996.

[42] M. A. Christodoulou, D. C. Theodoridis, and Y. S. Boutalis, "Building Optimal Fuzzy Dynamical Systems Description Based on Recurrent Neural Network Approximation", *in Proc. Int. Conf. of Networked Distributed Systems for Intelligent Sensing and Control*, Kalamata, Greece, June, 2007.

[43] B. Jakubezyk, "Feedback linearization of discrete time system", *Syst. Control Lett.*, vol. 9, pp. 411416, 1987.

[44] O. Adetona, S. Sathananthan, L.H. Keel, "Robust adaptive control of nonaffine nonlinear plants with small input signal changes", *IEEE Trans. on Neural Networks*, vol. 15, no. 2, pp. 408 - 416, March 2004. Nov. 2002.

[45] H.E. Psillakis, A.T. Alexandridis, "NN-Based Adaptive Tracking Control of Uncertain Nonlinear Systems Disturbed by Unknown Covariance Noise", *IEEE Trans. on Neural Networks*, vol. 18, no. 6, pp. 1830 - 1835, Nov. 2007.

[46] W. Leonhard, *Control of Electrical Drives*, Springer-Verlag, 1985.

[47] H. L. Hiew, C. P. Tsang, *Fuzzy Chaos and Recursive Partitioning*, in B. Kosko (Ed.), Fuzzy Engineering, Prentice-Hall International Inc. New Jersey, 1997.

[48] J. E. slotine, W. Li, *Applied Nonlinear Control*, Prentice-Hall International Inc. New Jersey, 1997.

[49] P. J. Antsaklis, K. M. Passino, *An Introduction to Intelligent and Autonomous Control*, Kluwer Academic Publishers, Norwell, MA, 1993.

[50] R. Ordonez, J. Zumberge, J. T. Spooner, K. M. Passino, "Adaptive Fuzzy Control: Experiments and Comparative Analyses", *IEEE Trans. on Fuzzy Systems*, vol. 5, no. 2, pp. 167 - 187, May. 1997.

[51] M. Maggiore, K.M. Passino, Output Feedback Tracking: A Separation Principle Approach, *IEEE Trans. on Automatic Control*, Vol. 50, No. 1, pp. 111-117, Jan. 2005.

[52] F. L. Lewis, S. Jagannathan, A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor and Fransis, Philadelphia, PA, 1999.