BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **5**, No 2

Sofia • 2005

A Recurrent Neural Multi-Model for Mechanical Systems Dynamics Compensation

Ieroham Baruch, Rafael Beltran*, Ruben Garrido*,*
*Boyka Nenkova***

** Department of Automatic Control, CINVESTAV-IPN, 07360 Mexico D.F., Mexico*
*** Institute of Information Technologies, 1113 Sofia*

Abstract: *The paper proposed a new fuzzy-neural recurrent multi-model for systems identification and states estimation of complex nonlinear mechanical plants with backlash. The parameters and states of the local recurrent neural network models are used for a local direct and indirect adaptive control systems design. The designed local control laws are coordinated by a fuzzy rule based control system. Simulation results confirm the applicability of the proposed intelligent control system, where a good convergence of all recurrent neural networks, is obtained.*

Keywords: *Recurrent neural networks, back propagation learning, fuzzy-neural multi-model, systems identification, adaptive control, mechanical system with backlash.*

1. Introduction

In the recent decade, the Neural Networks (NN) became universal tools for many applications. The NN modeling, and application to system identification, prediction, and control, was discussed for many authors [1-5]. Mainly, two types of NN models are used: Feedforward (FFNN) and Recurrent (RNN). The main problem here is the use of different NN mathematical descriptions, and control schemes, according to the structure of the plant model. For example, N a r e n d r a and P a r t h a s a r a t h y [5], applied FFNN for system identification and direct model reference adaptive control of various non-linear plants. They considered four plant models with a given structure and supposed that the order of the plant dynamics is known. Y i p and P a o [2], solved control and prediction problems by means of a flat-type functional FFNN used for inverse plant model learning control. P h a m and Y i l d i r i m, [3] applied Jordan RNN for robot control. S a s t r y and S a n t h a r a m, [4], introduced two types of neurons – Network Neurons and Memory Neurons to solve identification and adaptive control problems, considering that the plant model is also auto-regressive one. In [1], some schemes of NN and RNN applications to control, especially of direct model reference adaptive control, are surveyed. All drawbacks of the described in the literature NN models could be summarized as follows: 1) there exists a great variety of NN models and their universality is missing, [1-5]; 2) all NN models are sequential in nature, as implemented for systems identification (the FFNN model uses one or two tap-delays in the input, [5], and RNN models usually are based on the auto-regressive model, [1, 4], which is one-layer sequential one); the main drawback here is that the sequential models with different order introduces a different time delay in a parallel control scheme; 3) in more of the cases the stability of the RNN is not considered, [2, 5], especially during the learning; 4) in the case of FFNN application for systems identification, the plant is given in one of the four described in [5] plant models, the linear part of the plant model, especially the system order, has to be known and the FFNN approximates only the non-linear part of this model; 5) all these NN models are non-parametric ones, [4, 5] and so, not applicable for an adaptive control systems design; 6) all these models are appropriate for identification of nonlinear plants with smooth, single, odd nonsingular nonlinearities, [5].

B a r u c h et al., [6], in their previous paper, applied the state-space approach to describe RNN in an universal way, defining a Jordan canonical two– or three-layer RNN model, named Recurrent Trainable Neural Network (RTNN). This NN model is a system parameter and state estimator, which permits to use the obtained system states and parameters directly for state-space control.

For a complex nonlinear plant, B a r u c h, et al., [7, 9], proposed to apply a fuzzy-neural multi-model, appropriate to use when the nonlinear function in the control part of the plant is not invertible. Further, the proposed neural fuzzy-neural multimodel has been applied for mechanical system with friction identification, [10, 11]. In [12] the proposed multimodel approach has been used for an experimental DC motor identification. Latter, few control methods, using the fuzzy-neural multimodel, has been applied for mechanical plant with friction identification and control, [13-15]. Finally, in the last year, the results of some simulation and experimental work with a DC motor fuzzy-neural-multimodel control, has been presented, [16, 17].

In [18], a wide scope of references using fuzzy-neural approach for nonlinear plants approximation is given and the RNN architecture of Frasconi-Gori-Soda [19], is used. The main disadvantage of this work is that the applied RNN model there is sequential in nature. Depending on the model order, this RNN model generates different computational time delays, which makes difficult the fuzzy system synchronization, [18].

So, the aim of this paper is to go ahead, using a fuzzy-neural multimodel for iden-tification and control of nonlinear mechanical plants with backlash. The present paper proposes two adaptive neural multi-model control schemes – direct and indirect, illustrated by simulation results, obtained with a DC motor mechanical system with backlash.

2. Models description

2.1. Recurrent neural model and learning

A discrete-time model of Recurrent Trainable Neural Network (RTNN) and a dynamic Back propagation (BP) weight updating rule are given. The RTNN model is described by the following equations, [6]:

(1) $$X(k+1) = JX(k)+BU(k),$$

(2) $$Z(k)=S[X(k)],$$

(3) $$Y(k) = S[CZ(k)],$$

(4) $$J = \text{block-diag}\ (J_i);\ |J_i| < 1,$$

where $X(k)$ is an N-state vector of the system; $U(k)$ is an M-input vector; $Y(k)$ is an L-output vector; $Z(k)$ is an auxiliary vector variable with dimension L; $S(x)$ is a vector valued activation function with appropriate dimension; J is a weight-state diagonal matrix with elements J_i; B and C are weight input and output matrices with appropriate dimensions and block structure, corresponding to the block structure of J. The controlability, observability, and stability of this model are considered in [6]. As it can be seen, the given RTNN model is a completely parallel parametric one, so it is useful for identification and control purposes. Parameters of that model are the weight matrices J, B, C and the state vector $X(k)$. The equation (4) is a stability preserving condition. The general BP learning algorithm is given as:

(5) $$W_{ij}(k+1) = W_{ij}(k) + \eta\Delta W_{ij}(k) + \alpha\Delta W_{ij}(k-1),$$

where $W_{ij}(C, J, B)$ is the ij-th weight element of each weight matrix (given in parenthesis) of the RTNN model to be updated; ΔW_{ij} is the weight correction of W_{ij}; η, α are learning rate parameters. The updates ΔC_{ij}, ΔJ_{ij}, ΔB_{ij} of model weights C_{ij}, J_{ij}, B_{ij} are given by:

(6) $$\Delta C_{ij}(k) = [T_j(k) - Y_j(k)]\ S_j'\ (Y_j(k))\ Z_i(k),$$

(7) $$\Delta J_{ij}(k) = R\ X_i(k-1),$$

(8) $$R = C_i(k)\ [T(k) - Y(k)]\ S_j'\ (Z_j(k)),$$

(9) $$\Delta B_{ij}(k) = R\ U_i(k),$$

where T is a target vector with dimension L and $[T-Y]$ is an output error vector also with the same dimension; R is an auxiliary variable; $S'(x)$ is the derivative of the activation function, which for the hyperbolic tangent is $S_j'(x) = 1-x_2$. The stability of the learning algorithm and its applicability for systems identification and control, are proven in [6], where a DC motor is controlled by a direct adaptive neural control system, containing a neural identifier and a neural controller.

2.2. Fuzzy-neural multi-model

For a complex dynamic systems identification, the Takagi–Sugeno fuzzy rule, [20], admits to use in the consequent part a crisp function, which could be a static or dynamic (state-space) model, [1, 20, 21], which validation is determined by the membership function. Some authors, referred in [18], proposed as a consequent crisp function to use a *NN* function. B a r u c h et al. [7-11], proposed as a consequent crisp function to use a RTNN function model, so to form a fuzzy-neural multi-model. The following statement gives the fuzzy rule of the model:

(10) R_i: if x is A_i then $y_i(k+1) = N_i\,[x(k), u(k)]$, $i = 1, 2, \ldots, P$,

where N_i (.) denotes the RTNN function, given by equations (1) to (3); i is the model number; P is the total number of RTNN models, corresponding to fuzzy rules R_i. The output of the fuzzy neural multi-model system is given by the following equation

(11) $Y = \Sigma_i\, w_i\, y_i = \Sigma_i\, w_i\, N_i(x, u)$,

where w_i are weights, obtained from the membership functions, [13-15]. As it could be seen from the equation (11), the output of the fuzzy-neural multi model, which approximates the nonlinear plant model could be obtained as a weighted sum of RTNN models, [7-17], given in the consequent part of (10). The weights of the neural model could be learned, which is the great advantage of this neural multi-model. In the case when the intervals of the variables given in the antecedent parts of the rules are not overlapping, the weights obtain values one and the weighted sum (11) is converted in a simple sum. This simple particular case, considered here, was called fuzzy-neural multi-model, [7-17].

3. An adaptive fuzzy-neural control systems design

3.1. A direct adaptive fuzzy-neural control

Block-diagram of a direct adaptive fuzzy-neural multi model control system is given in Fig. 1. The block-diagram contains one fuzzy-neural multi-model identifier, which issued states to the fuzzy-neural multi-model controller. The structure of the entire identification system contains a fuzzyfier, a Fuzzy Rule-Based System (FRBS), and a set of RTNN models. The system does not need a defuzzyfier, because the RTNN models are crisp limited state-space models. A possible adaptive control system contains also a set of RTNN controllers incorporated in a FRBS, designed on the base of the obtained set of RTNN's.

$U_i(k) = -Nf_{b,i}\,[x_i(k)] + Nf_{f,i}\,[r_i(k)]$,

(12) R_i: if x is A_i Then $u_i = U_i(k)$, i=1, 2 , ..., P,

where $r(k)$ is the reference signal; $x(k)$ is the system state; $N_{FBi}\,[x_i(k)]$ and $N_{FFi}\,[r_i(k)]$ are the feedback (FB) and feedforward (FF) parts of the fuzzy-neural control. The control issued by the fuzzy neural multi-model system is given by the following equation:

(13) $U(k) = \Sigma_i\, w_i\, U_i\,(k)$,

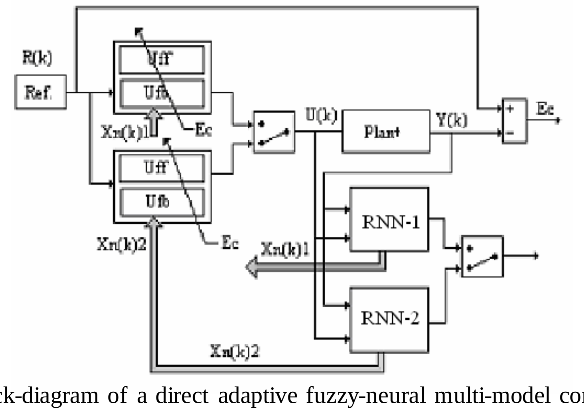

Fig. 1. Block-diagram of a direct adaptive fuzzy-neural multi-model control system

where w_i are weights, obtained from the membership functions, [7, 17, 18], corresponding to the rules (13). As it could be seen from the equation (13), the control could be obtained as a weighted sum of controls, given in the consequent part of (12). In the case when the intervals of the variables, given in the antecedent parts of the rules, are not overlapping, the weights obtain values one and the weighted sum (13) is converted in a simple sum. For sake of simplicity, this particular model, named multi-model, is considered here.

3.2. An indirect adaptive fuzzy-neural control

The block diagram of the indirect adaptive fuzzy-neural control system is given in Fig. 2. The structure of the entire identification system is the same as in the previous control scheme, but here a linear control law is designed using the obtained state ($x_i(k)$) and parameter (J_i, B_i, C_i) information, issued by the local neural model identifiers RTNN-1,2. The multi model control is given by the same equation (13), consequence of the rule (12) application, but here the local control (see [13]), is given by:

(14) $$U_i(k) = (C_i B_i) -1\{C_i J_i X_i(k) + r_i(k+1) + \gamma [r_i(k) - Y_i(k)]\}.$$

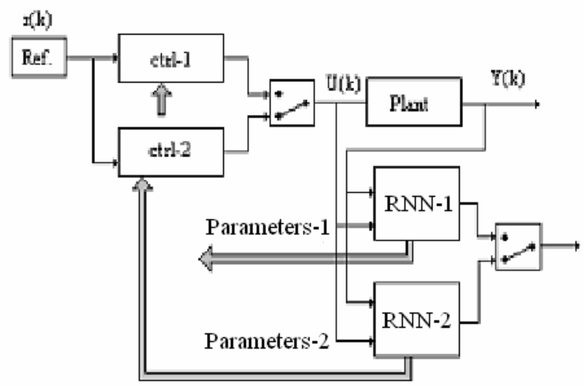

Fig. 2. Block-diagram of the indirect adaptive fuzzy-neural multi-model control system

In this particular case, we use only two neural nets for process identification. The RTNN-1 corresponds to the positive part of the plant output signal, and the RTNN-2 corresponds to the negative one. For this two neural models – two correspondent controls $U_1(k)$ and $U_2(k)$ are computed using (14), where the value of the control parameter γ is chosen between –0.999 and 0.999, and $r_i(k)$ is the correspondent local reference signal (also positive or negative). If the RTNN-i model is observable and controllable, then the local matrix product $C_i B_i$ is different from zero ($C_i B_i \neq 0$).

4. Simulation results

Let us consider an electromechanical system, driven by a DC motor, [22-26], governed by the equations:

(15) $$K_b \omega_m(t) = R_a i_a + L_a \frac{di_a(t)}{dt} + e_a \,,$$

$$K_i i_a(t) = J_m \frac{d^2\theta_m(t)}{dt} + b_m \frac{d\theta_m(t)}{dt} + \tag{16}$$
$$+ \lambda^2 J_L \frac{d^2\theta_L(t)}{dt} + \lambda^2 T_L; \lambda = \left(\frac{N_1}{N_2}\right),$$

where $i_a(t)$ is the armature current; $e_a(t)$ is the applied voltage; R_a, L_a are the armature resistance and inductance; J_m, J_L are the rotor and load inertias; $\omega_m(t)$ is the angular velocity of the rotor; $\theta_m(t)$, $\theta_L(t)$ are the rotor and load angular positions; K_b, K_i are electromechanical and electrical constants; bm is a viscose friction coefficient; *TL* is the moment of load; λ is the gear ratio; N_1, N_2 are gear numbers. The backlash model, [23-26], is illustrated in Fig. 3. The backlash characteristic $\theta_L(t) = B(\theta_m(t)) = B(m, C_R, C_L, \theta_m(t))$ is described by two parallel lines, connected by horizontal line segments. The following up direction movement is active when both input $\theta m(t)$ and output $\theta_L(t)$ angular positions are incremented, e.g.

$$\theta_L(t) = m(\theta_m(t) - C_R),\ \theta_m(t) > 0,\ \theta_L(t) > 0. \tag{17}$$

The following opposite down direction movement is active when both input $\theta_m(t)$ and output $\theta_L(t)$ angular positions are decremented, e.g.

$$\theta_L(t) = m(\theta_m(t) - C_L),\ \theta_m(t) < 0,\ \theta_L(t) < 0, \tag{18}$$

where $m > 0$, $C_L < C_R$ are constant parameters. The gear backlash model is taken from the literature [23-26]. The *DC* motor driven electromechanical system governed by equations (15), (16), together with the gear backlash equations (17) and (18), are simulated using Matlab/Simulink TM version 6.1 software, and its output variables are discretized by a sufficiently small period of discretization, taking into account the Shannon theorem. The control signal is retained by zero hold. The DC motor and backlash parameters used, are given in Table 1.

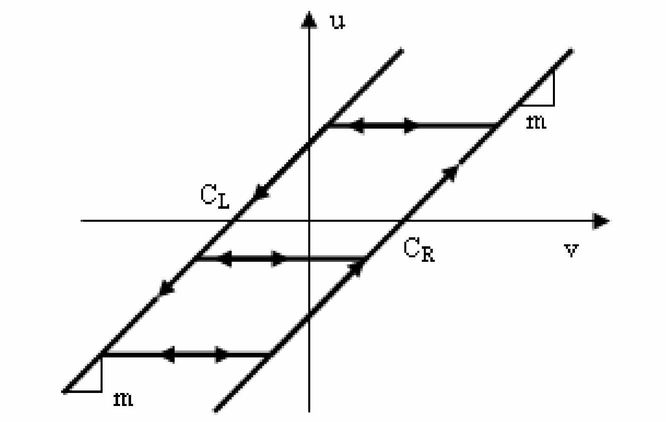

Fig. 3. Backlash model, where: $v(t)$ is the input and $u(t)$ is the output; $C_R > 0$ is the right path and $C_L < 0$ is the left path

Table 1. DC Motor and backlash parameters

Parameter	Value
L_a – inductance	0.055 H
R_a – resistance	7.56 Ω
K_b – constant	3.475 N.m.A^{-1}
J_m – inertia of the DC-motor rotor	0.068 kg.m^2
B_m–coefficient of viscose friction	0.03475 N.m.s
m – mass	1 kg
CR–backlash constant of the right pass	0.2 mm
CL – backlash constant of the left pass	–0.2 mm

Simulation results obtained by means of the direct neural multimodel control scheme are given on Figs. 4, a-e. The total time of learning is 500 s. The identification and control RTNNs topologies are (1, 5, 1), (5, 5, 1) and (1, 5, 1), respectively. The learning parameters are given in Table 2.

Table 2. Learning parameters

Parameter	Value
η – learning rate constant	0.01
α – momentum term constant	0.01
T_m – period of discretization	0.01 s

The reference signal is given by the following equation:

(19) $$r(k) = 10\sin(\omega_k); \quad \omega = 1 \text{ rad/s}.$$

From the graphical results, shown in Figs. 4, a-e it is seen that all neural networks are convergent and the output of the plant follows the reference signal in spite of the backlash and the Means Squared Error of reference tracking is below 1%. Simulation results obtained by means of the indirect neural multi-model control scheme are given on Figs. 5, a-d. The learning and control parameters are given in Table 3. The reference signal is given by (19). The total learning time is 120 s. The topology of the neural identifier is (1, 5, 1).

Table 3. Learning and control parameters

Parameter	Value
η – learning rate constant	0.001
α – momentum term constant	0.01
T_m – period of discretization	0.01 s
γ – control parameter	0.5

From the graphical results, shown in Figs. 4-5, it is seen that the identification neural networks are convergent and the output of the plant follows the reference signal in spite of the backlash and the Means Squared Error of reference tracking is below 2.5%.

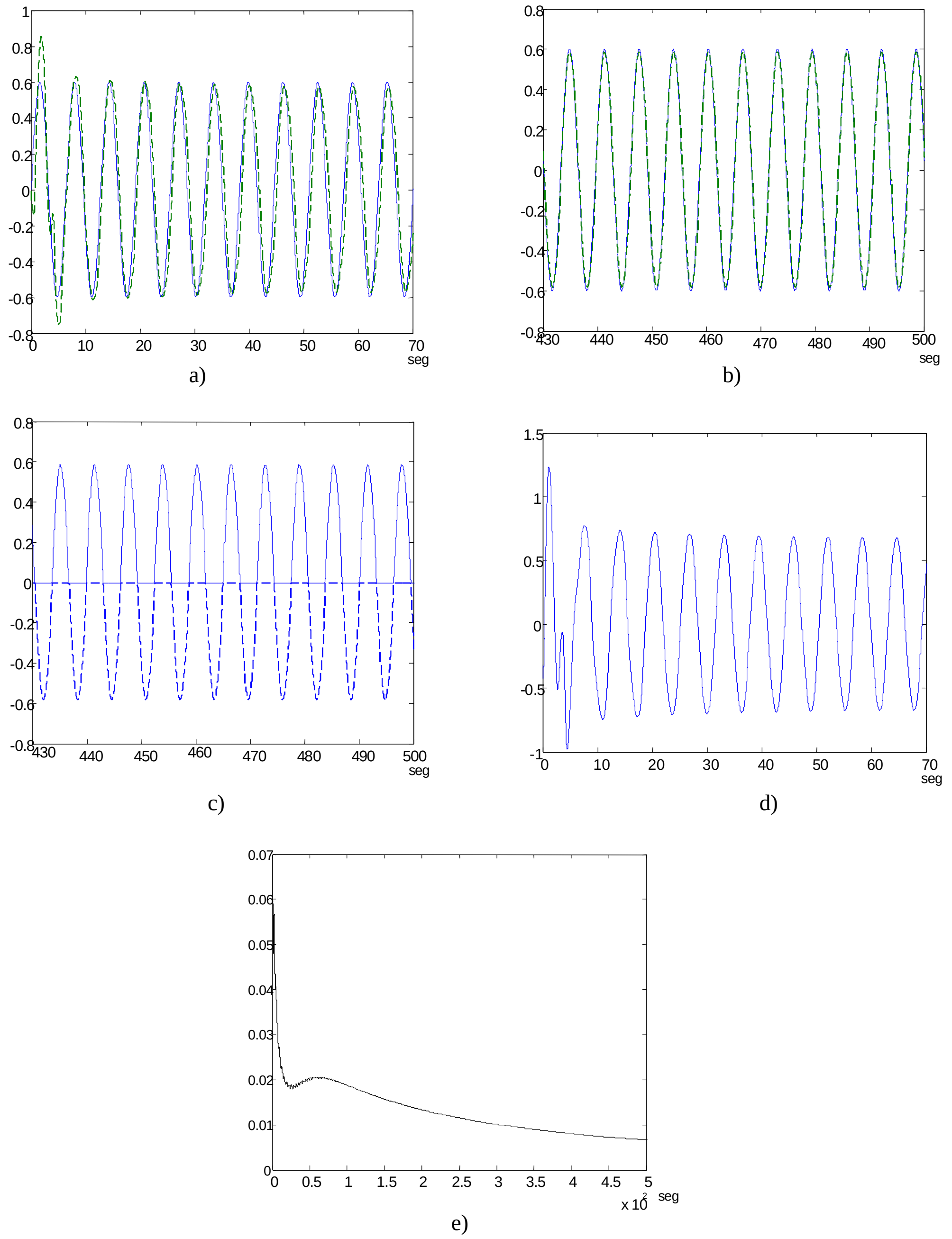

Fig. 4. Graphical results of the direct adaptive neural multi-model control for an electromechanical plant with backlash: a) – comparison of the reference signal (continuous line), and the plant output (dashed line), during first 70 s of simulation; b) – comparison of the reference signal (continuous line), and the plant output (dashed line), during last 70 s of simulation; c) – graphical results of the closed-loop systems identification. Graphics of the RTNN-1 output (continuous line), and the RTNN-2 output (dashed line), during last 70 s of learning; d) – control signal during first 70 s of simulation; e) – Mean Squared Error (MSE%) of control

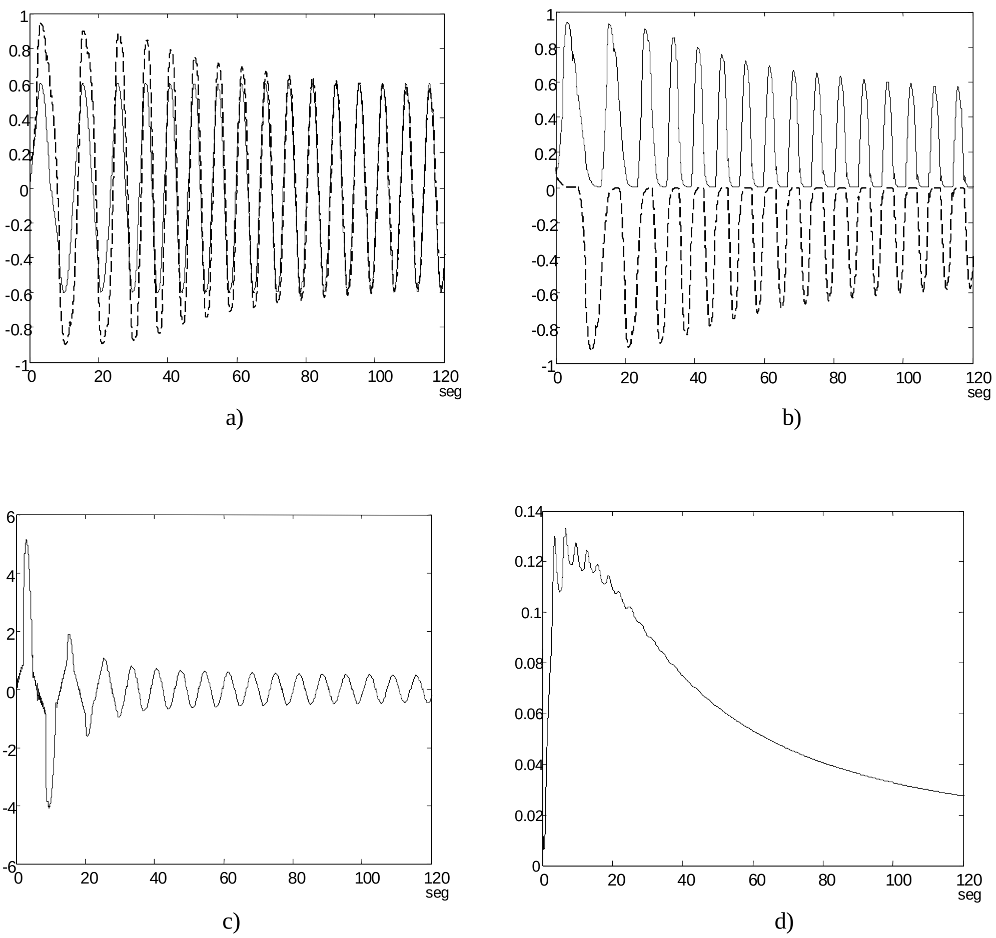

Fig. 5. Graphical results of the indirect adaptive neural multi-model control for an electromechanical plant with backlash: a) – comparison of the reference signal (continuous line), and the plant output (dashed line), during 120 s of simulation; b) – graphical results of the closed-loop systems identification. Graphics of the RTNN-1 output (continuous line), and the RTNN-2 output (dashed line), during 120 s of learning: c) – control signal during 120 s of simulation; d) – Mean Squared Error (MSE%) of control

5. Conclusions

A two-layer Recurrent Neural Network (RNN) and an improved dynamic error Backpropagation- method of its learning, are described. For a complex nonlinear plant identification and control, a fuzzy-neural multi-model, is used. The fuzzy-neural multi-model, containing two RNNs, is applied for real-time identification and adaptive direct and indirect control of nonlinear mechanical system with gear backlash, where the simulation results exhibits a good convergence. The obtained good simulation results confirm the applicability of the proposed fuzzy neural multi-model control scheme.

References

1. H u n t, K. J., D. S b a r b a r o, R. Z b i k o w s k i, P. J. G a w t h r o p. Neural network for control systems – a survey. – Automatica, **28**, 1992, 1083-1112.
2. Y i p, P. P. C., Y. H. P a o. A recurrent neural net approach to one-step ahead control problems.– IEEE Trans. Syst., Man, Cybern., **24**, 1994, 678-683.
3. P h a m, D. T., S. Y i l d i r i m. Robot control using Jordan neural networks. – In: Proc. of the International Conference on Recent Advances in Mechatronics, August 14-16, 1995, Istanbul, Turkey, Vol. II, 888-893.
4. S a s t r y, P. S., G. S a n t h a r a m, K. P. U n n i k r i s h n a n. Memory networks for identification and control of dynamical systems. – IEEE Transactions on NNs, **5**, 1994, 306-320.
5. N a r e n d r a, K. S., K. P a r t h a s a r a t h y. Identification and control of dynamic systems using neural networks. – IEEE Transactions on NNs, **1**, 1990, 4-27.
6. B a r u c h, I., J. M. F l o r e s, F. N a v a, I. R. R a m i r e z, B. N e n k o v a. An advanced neural network topology and learning, applied for identification and control of a DC Motor. – In: Proc. of the First International IEEE Symposium on Intelligent Systems, IS'02, Varna, Bulgaria, September 10-12, 2002, Vol. I, 289-295.
7. B a r u c h, I., E. G o r t c h e v a. Fuzzy-neural model for nonlinear systems identification. – In: Proc. of the 5th IFAC Workshop AARTC'98, Cancun, Mexico, April 15-17, 1998, 283-288.
8. B a r u c h, I., T. A r s e n o v, E. G o r t c h e v a, R. G a r r i d o. A fuzzy-neural model for dynamic systems identification and an intelligent control. – In: Proc. of the 5th International Symposium on "Methods and Models in Automation and Robotics", Miedzyzdroje, Poland, 25-29 August., 1998, Vol. 2, 624-630.
9. B a r u c h, I., E. G o r t c h e v a, F. T h o m a s, R. G a r r i d o. A neuro-fuzzy model for nonlinear plants identification. – In: Proc. of the IASTED Int. Conference on Modeling and Simulation, MS?99, Philadelphia, PA, USA, May 5-8, 1999, 1-6.
10. B a r u c h, I., R. G a r r i d o, A. M i t e v, B. N e n k o v a. A neural network approach for stick-slip model identification. – In: Proc. of the 5th International Conference on Engineering Applications of Neural Networks, EANN'99, Warsaw, Poland, September 13-15, 1999, 183-188.
11. B a r u ch, I., F. T h o m a s, R. G a r r id o, E. G o r t c h e v a. A hybrid multimodel neural network for nonlinear systems identification. – In: Int. Joint Conference on Neural Networks, IJCNN'99, Washington D.C., USA, July 10-16, 1999, Vol. 6, 4278-4283.
12. B a r u c h, I., J. M. F l o r e s, J. C. M a r t i n e z, R. G a r r i d o. A multi-model parameter and state estimation of mechanical systems. – In: Proc. of the IEEE International Symposium on Industrial Electronics, ISIE'2000, 4-8 December 2000, Puebla, Mexico, Vol. 2, 700-705.
13. B a r u c h, I., J. M. F l o r e s, J. C. M a r t i n e z, B. N e n k o v a. Fuzzy- neural models for real-time identification and control of a mechanical system. (St. Cerri, D. Dochev Eds.) – In: Artificial Intelligence: Methodology, Systems and Applications. – In: Proc. of the 9th International Conference, AIMSA 2000, September 2000, Varna, Bulgaria, LNAI 1904, Berlin, Heidelberg, New York, Springer Verlag, 292-300.
14. B a r u c h, I., J. M. F l o r e s, R. G a r r i d o. A fuzzy-neural recurrent multimodel for systems identification and control. – In: Proc. of the European Control Conference, ECC'01, Porto, Portugal, September 4-7, 2001, 3540-3545.
15. B a r u c h, I., J. M. F l o r e s, F. T h o m a s, E. G o r t c h e v a. A multimodel recurrent neural network for systems identification and control. – In: Proc. of the International Joint Conference on Neural Networks, IJCNN'01, Washington D.C., USA, July 14-19, 2001, 1291-1296.
16. B a r u c h, I., R. B e l t r a n, J. L. O l i v a r e s, R. G a r r i d o. A fuzzy-neural multi-model for mechanical systems identification and control. (Raul Monroy, Gustavo Arroyo-Figueroa, Luis Enrique Sucar, Humberto Sossa, Eds.) – In: Advances in Artificial Intelligence, MICAI 2004, Proc. of the 3th Mexican International Conference on Artificial Intelligence, Mexico City, Mexico, April 2004, LNAI 2972, Berlin, Heidelberg, New York, Springer Verlag, 2004, 774-783.
17. B a r u c h, I., R. B e l t r a n, B. N e n k o v a. A mechanical system backlash compensation by means of a recurrent neural multi-model. – In: Proc. of the 2nd International IEEE Conference on Intelligent Systems, Varna, Bulgaria, June 22-24, 2004, Vol. II, 514-519.

18. M a s t o r o c o s t a s, P. A., J. B. T h e o c h a r i s. A recurrent fuzzy-neural model for dynamic system identification. – IEEE Trans. on Syst., Man, Cybern. Part B: Cybernetics, **32**, 2002, 176-190.
19. F r a s c o n i, P., M. G o r i, G. S o d a. Local feedback multilayered networks. – Neural Computation, **4,** 1992, 120-130.
20. T a k a g i, T. M., S u g e n o. Fuzzy identification of systems and its applications to modeling and control. – IEEE Trans. Syst., Man, Cybern., **15**, 1985, 116-132.
21. T e i x e i r a, M., S. Z a k. Stabilizing controller design for uncertain nonlinear systems using fuzzy models. – IEEE Trans. Syst., Man, Cybern., **7**, 1999, 133-142.
22. W e e r a s o o r i y a, S. M., A. E l-S h a r k a w i. Identification and control of a DC-motor using backpropagation neural networks. – IEEE Trans. on Energy Conversion, **6**, 1991, 663-669.
23. T a o, G a n g, P. V. K o k o t o v i c. Adaptive Control of Systems with Actuator and Sensor Nonlinearities. London, John Wiley and Sons Inc., 1996.
24. C i n c o t t i, S., L. D a n e r i. Neural network identification of a nonlinear circuit model of hysteresis. – Electronic Letters, **33**, 1997, 1154-1156.
25. M e n o n, K., K. K r i h n a m u r t h y. Control of low velocity friction and gear backlash in a machine tool feed drive system. – Mechatronics, **9**, 1999, 33-52.