Sliding Mode Feedback-Error-Learning Neurocontrol Strategy for a Robot Manipulator

Andon V. Topalov[1] and Okyay Kaynak
Department of Electrical and Electronic Engineering, Mechatronics Research and Application Center, Bogazici University, Bebek, 34342 Istanbul, Turkey
e-mail: topalov@boun.edu.tr, kaynak@boun.edu.tr

***Abstract*—The features of a novel dynamical algorithm for robust adaptive learning in neural networks and its application to the neuro-adaptive nonlinear feedback control are presented. The proposed approach makes a direct use of variable structure systems theory. The suggested learning algorithm establishes an inner sliding motion in terms of the controller parameters, leading the learning error toward zero. The outer sliding motion concerns the controlled nonlinear plant, the state tracking error vector of which is simultaneously forced towards the origin of the phase space. The equivalence between the two sliding motions is demonstrated. The convergence of the algorithm is established and conditions are given. Results from a simulated neuro-adaptive control of a two-link planar manipulator by using the proposed nonlinear regulator sliding mode feedback-error-learning scheme are presented. They show that the implemented neural controller inherits some of the advantages of the variable structure systems: high speed of learning and robustness.**

I. Introduction

THE applications of neural networks in closed-loop feedback control systems have only recently been rigorously studied. As it is pointed in [1], when placed in a feedback system, even a static neural network becomes a dynamical system and takes on new and unexpected behaviors. Hence, such properties of the neural structures as their internal stability, passivity, and robustness must be studied before conclusions about the closed-loop performance can be made. Many early papers on neurocontrol did not address these effects. They mainly proposed some control topologies and used some standard techniques for weight adaptation.

Variable structure systems (VSS) with sliding mode were first proposed in the early 1950s. The best property of the sliding mode control (SMC) scheme is its robustness. Broadly speaking, a system with an SMC is insensitive to parameter changes or external disturbances [2]. Recent studies have emphasized that the convergence properties of the gradient-based training strategies widely used in artificial neural networks can be improved by utilizing the SMC approach [3]. The method presented in [3] can be considered as an indirect use of VSS theory. Some studies on the direct use of SMC strategy have been also reported in the literature [4] – [6]. A sliding mode strategy for adaptive learning in Adalines has been developed in [4] and algorithms for training of multilayer perceptrons, by defining separate sliding surfaces for the different neuron layers, have been proposed in [5, 6]. Their applications in neuro-adaptive control schemes have been investigated in [7] – [9].

Robotic manipulators are frequently used as test bed for evaluation of newly proposed computationally intelligent control methods since their coupled nonlinear equations and ambiguities in the friction related dynamics inevitably require flexible control architectures. The use of neural networks (NNs) for learning manipulator inverse dynamics has been a common approach. Jordan has proposed *forward-inverse-modeling* [10] and Albus [11], Miller et al. [12] and other researchers have used *direct-inverse-modeling* to obtain command-error for forming the inverse dynamics model as a feedforward neurocontroller. In *reinforcement learning* [13] the plant performance can be improved over time by means of on-line learning methods in less structured situations. The *feedback-error-learning* approach, proposed by Kawato et al. [14], is based on the NN realization of the computed torque, plus a secondary proportional plus derivative (PD) controller. The output of the controller has been used as an error signal to update the weights of a NN trained to become a feedforward controller. Subsequently Gomi and Kawato [15] have extended it for learning schemes where NN is applied as an adaptive nonlinear feedback controller.

In the present paper the feedback-error-learning control approach is further investigated by applying a VSS-based on-line learning algorithm to the NN feedback controller (NNFC). An inner sliding motion in terms of the NNFC parameters is established, leading the output signal of the

[1] On leave from the Department of Control Systems, Technical University of Sofia – Plovdiv branch, 24, Canko Dustabanov str., 4000 Plovdiv, Bulgaria

conventional feedback controller (CFC) toward zero. The outer sliding motion concerns the system under control, the state tracking error vector of which is simultaneously forced towards the origin of the phase space.

Section II presents the VSS-based on-line learning algorithm, the proposed sliding mode feedback-error-learning control scheme and introduces the equivalency constraints on the sliding control performance for the plant and the learning performance for the NNFC. Results from a simulated control of a two-link planar manipulator by using this neurocontrol strategy are shown in section III. Finally, section IV summarizes the findings of the work.

II. The Sliding Mode Learning Algorithm

A. Initial assumptions and definitions

Consider the two-layered feedforward NN shown on Fig. 1. We will use the following definitions:

$X(t) = \left[x_1(t), \ \ldots \ , x_p(t) \right]^T$ - vector of the time-varying input signals augmented by the bias term.

$\mathrm{T}_H(t) = \left[\tau_{H1}^n(t), \ \ldots \ , \tau_{Hn}^n(t) \right]^T$ - vector of the output signals of the neurons in the hidden layer.

$\tau^n(t)$ - scalar signal representing the time-varying output of the network.

$W1(t)_{(n \times p)}$ - matrix of the time-varying connections' weights between the neurons in the input and the hidden layer, where each matrix's element $w1_{i,j}(t)$ means the weight of the connection of the neuron i from its input j.

$W2(t)_{(1 \times n)} = \left[w2_1(t), \ldots, w2_n(t) \right]$ - vector of the time-varying connections' weights between the neurons in the hidden layer and the output node. Both $W1(t)_{(n \times p)}$ and $W2(t)_{(1 \times n)}$ are considered augmented by including the bias weight components for the neurons in the hidden layer and the output neuron respectively.

$f(\cdot)$ - nonlinear, differentiable, monotonously increasing activation function of the neurons in the hidden layer of the network. The neuron in the output layer is considered with a linear activation function.

The NNFC is assumed to operate within an adaptive control scheme, the general structure of which is presented in Fig. 2. A conventional feedback controller (CFC) is provided both as an ordinary feedback controller to guarantee global asymptotic stability in compact space and as an inverse reference model of the response of the controlled object.

The proposed learning scheme has three main differences from the one presented earlier in [15].

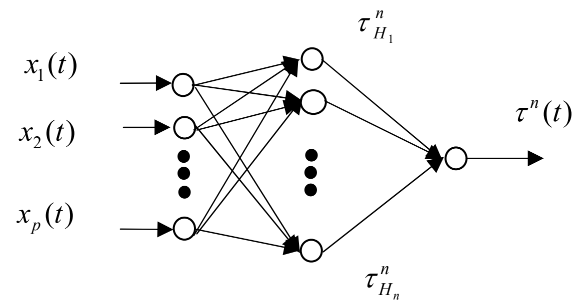


Fig. 1. Multilayer feedforward neural network with a scalar output

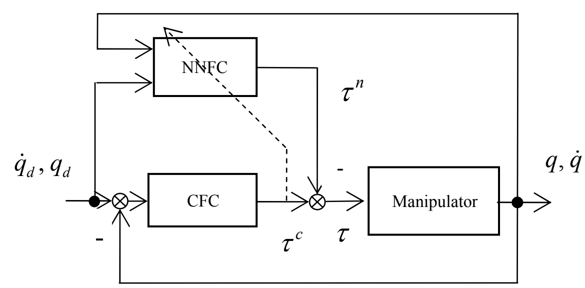


Fig. 2. Block diagram of nonlinear regulator sliding mode feedback-error-learning scheme.

- The equation of the system is

$$\tau = \tau^c - \tau^n \tag{1}$$

- A PD control law is used in CFC:

$$\tau^c = k_D \dot{e} + k_P e \tag{2}$$

where k_D and k_P are the controller gains.

- Sliding mode feedback-error-learning algorithm is applied to the NNFC. The sliding surface for the controlled system $s_p\left(e, \dot{e}\right)$ and the zero adaptive learning error level for the NNFC $s_c\left(\tau, \tau^n\right)$ are defined as follows:

$$s_p\left(e, \dot{e}\right) = \dot{e} + \lambda e \tag{3}$$

$$s_c\left(\tau^n, \tau\right) = \tau^c = \tau^n + \tau \tag{4}$$

where λ determines the slope of the sliding surface.

An assumption is made that the input vector of the NNFC $X(t)$ and it's time derivative $\dot{X}(t)$ are bounded, i.e.

$$\left\| X(t) \right\| = \sqrt{x_1^2(t) + \ldots + x_p^2(t)} \le B_X \quad \forall t \tag{5}$$

$$\left\| \dot{X}(t) \right\| = \sqrt{\dot{x}_1^2(t) + \ldots + \dot{x}_p^2(t)} \le B_{\dot{X}} \quad \forall t$$

where B_X and $B_{\dot{X}}$ are known positive constants.

It will be accepted that, due to the physical constraints, the magnitude of all vectors row $W1_i(t)$ constituting the matrix $W1(t)$ and the elements of the vector $W2(t)$ are also bounded at each instant of time t by means of

$$\|W1_i(t)\| = \sqrt{w1_{i,1}^2(t) + w1_{i,2}^2(t) + ... + w1_{i,p}^2(t)} \leq B_{W1} \quad \forall t,$$
$$|w2_i(t)| \leq B_{W2} \quad \forall t \tag{6}$$

for some known constants B_{W1} and B_{W2}, where $i = 1, 2, ..., n$.

It will be also adopted that $\tau(t)$ and $\dot{\tau}(t)$ are also bounded signals, i.e.

$$|\tau(t)| \leq B_\tau, \ |\dot{\tau}(t)| \leq B_{\dot{\tau}} \quad \forall t \tag{7}$$

where B_τ and $B_{\dot{\tau}}$ are positive constants.

The output signal $\tau_{H_i}^n$ of the i-th neuron from the hidden layer and the output signal of the network $\tau^n(t)$ are defined as:

$$\tau_{H_i}^n = f\left(\sum_{j=1}^{p} w1_{i,j} x_j\right), \quad \tau^n(t) = \sum_{i=1}^{n} w2_i \tau_{H_i}^n \tag{8}$$

B. The SMC-based on-line learning algorithm

Using the SMC approach, the zero value of the learning error coordinate $\tau^c(t)$ is defined as a time-varying sliding surface, i.e.

$$s_c\left(\tau^n, \tau\right) = \tau^c(t) = \tau^n(t) + \tau(t) = 0 \tag{9}$$

which is the condition that the neural network model is trained to become a nonlinear regulator to obtain the desired response during tracking-error convergence movement by compensation for the nonlinearity of the controlled plant.

Definition 2.1: A sliding motion will have place on a sliding manifold $s_c\left(\tau^n, \tau\right) = \tau^c(t) = 0$, after time t_h (hitting time) if the condition $s_c(t)\dot{s}_c(t) = \tau^c(t)\dot{\tau}^c(t) < 0$ is true for all t in some nontrivial semi open subinterval of time of the form $\left[t, t_h\right) \subset \left(-\infty, t_h\right)$.

The learning algorithm for the network weights should be derived in such a way that the sliding mode condition of the above definition 2.1 will be enforced. To enable $s_c = 0$ is reached, we have the following theorem:

Theorem 2.2: If the learning algorithm for the weights $W1(t)$ and $W2(t)$ is chosen respectively as

$$\dot{w}1_{i,j} = -\left(\frac{w2_i x_j}{X^T X}\right)\alpha \ sign(s_c), \tag{10}$$
$$\dot{w}2_i = -\left(\frac{\tau_{H_i}^n}{\mathrm{T}_H^T \mathrm{T}_H}\right)\alpha \ sign(s_c)$$

with α being sufficiently large positive constant satisfying

$$\alpha > nB_A B_{W1} B_{\dot{X}} B_{W2} + B_{\dot{\tau}} \tag{11}$$

then, for any arbitrary initial condition $s_c(0)$, the learning error $\tau^c(t)$ will converge to zero during a finite time t_h which may be estimated as

$$t_h \leq \frac{|s_c(0)|}{\alpha - nB_A B_{W2} B_{W1} B_{\dot{X}} + B_{\dot{\tau}}} \tag{12}$$

and a sliding motion will occur on $\tau^c = 0$ for all $t > t_h$.

Proof 2.2: Consider the following Lyapunov function candidate:

$$V_c = \frac{1}{2} s_c^2 \tag{13}$$

Then differentiating V_c yields

$$\dot{V}_c = s_c \dot{s}_c = s_c\left(\dot{\tau}^n + \dot{\tau}\right) = \tag{14}$$
$$s_c\left\{\left[\sum_{i=1}^{n} w2_i f\left(\sum_{j=1}^{p} w1_{i.j} x_j\right)\right]' + \dot{\tau}\right\} =$$
$$s_c\left[\sum_{i=1}^{n} \dot{w}2_i \tau_{H_i}^n + \sum_{i=1}^{n} w2_i A_i \sum_{j=1}^{p}\left(\dot{w}1_{i,j} x_j + w1_{i,j}\dot{x}_j\right) + \dot{\tau}\right] =$$
$$s_c\left[-\sum_{i=1}^{n} \frac{\tau_{H_i}^n}{\mathrm{T}_H^T \mathrm{T}_H}\alpha \ sign(e)\tau_{H_i}^n + \right.$$
$$\left. + \sum_{i=1}^{n} A_i \sum_{j=1}^{p}\left(-\left(\frac{w2_i x_j}{X^T X}\right)\alpha \ sign(e) x_j w2_i + w1_{i,j}\dot{x}_j w2_i\right) + \dot{\tau}\right] =$$
$$s_c\left(-\alpha sign(s_c) - \sum_{i=1}^{n} A_i \alpha w2_i^2 sign(s_c) + \sum_{i=1}^{n} A_i w2_i \sum_{j=1}^{p} w1_{i,j}\dot{x}_j + \dot{\tau}\right) =$$
$$-\left(\alpha + \alpha\sum_{i=1}^{n} A_i w2_i^2\right)|s_c| + \left(\sum_{i=1}^{n} A_i w2_i \sum_{j=1}^{p} w1_{i,j}\dot{x}_j + \dot{\tau}\right)s_c \leq$$
$$-\alpha|s_c| + s_c\left(\sum_{i=1}^{n} A_i w2_i \sum_{j=1}^{p} w1_{i,j}\dot{x}_j + \dot{\tau}\right) \leq$$

$$-\alpha|s_c| + \left(nB_A B_{W2} B_{W1} B_{\dot{X}} + B_{\dot{\tau}}\right)|s_c| =$$
$$|s_c|\left(-\alpha + nB_A B_{W2} B_{W1} B_{\dot{X}} + B_{\dot{\tau}}\right) < 0 \quad \forall s_c \neq 0$$

where $A_i(t)$, $0 < A_i(t) = f'\left(\sum_{j=1}^{p} w1_{i,j} x_j\right) \leq B_A \quad \forall i, j$ is the derivative of the neurons' activation function $f(.)$ and B_A corresponds to its maximum value. The inequality (14) means that the controlled trajectories of the learning error $s_c(t)$ converge to zero in a stable manner.

It is possible now to be shown that such a convergence takes place in finite time. The differential equation that is satisfied by the controlled error trajectories $s_c(t)$ is as follows

$$\dot{s}_c(t) = -\left(1 + \sum_{i=1}^{n} A_i w2_i^2\right)\alpha\, sign(s_c) + \sum_{i=1}^{n} A_i w2_i \sum_{j=1}^{p} w1_{i,j} \dot{x}_j + \dot{\tau} \quad (15)$$

For any $t \leq t_h$, the solution, $s_c(t)$, of this equation, with initial condition $s_c(0)$ at $t = 0$, satisfies

$$s_c(t) - s_c(0) = \int_0^t \dot{s}_c(\sigma) d\sigma = \quad (16)$$
$$\int_0^t \left[-\alpha sign(s_c(\sigma))\left(1 + \sum_{i=1}^{n} A_i(\sigma) w2_i^2(\sigma)\right) + \right.$$
$$\left. + \sum_{i=1}^{n} A_i(\sigma) w2_i(\sigma) \sum_{j=1}^{p} w1_{i,j}(\sigma)\dot{x}_j(\sigma) + \dot{\tau}(\sigma)\right] d\sigma$$

At time $t = t_h$ the solution takes zero value and, therefore,

$$-s_c(0) = -\alpha\, sign(s_c(0))\left[t_h + \int_0^{t_h}\left(\sum_{i=1}^{n} A_i(t) w2_i^2(t)\right)dt\right] +$$
$$+ \int_0^{t_h}\left(\sum_{i=1}^{n} A_i(t) w2_i(t) \sum_{j=1}^{p} w1_{i,j}(t)\dot{x}_j(t) + \dot{\tau}(t)\right)dt \quad (17)$$

By multiplying both sides of the equation with $-sign(s_c(0))$ the estimate of t_h in (12) can be found using the inequality (18).

$$|s_c(0)| = \alpha t_h + \alpha \int_0^{t_h}\left(\sum_{i=1}^{n} A_i(t) w2_i^2(t)\right)dt - \quad (18)$$
$$sign(s_c(0)) \int_0^{t_h}\left(\sum_{i=1}^{n} A_i(t) w2_i(t) \sum_{j=1}^{p} w1_{i,j}(t)\dot{x}_j(t) + \dot{\tau}(t)\right)dt \geq$$
$$\alpha\left(t_h + \int_0^{t_h}\left(\sum_{i=1}^{n} A_i(t) w2_i^2(t)\right)dt\right) - \left(nB_A B_{W2} B_{W1} B_{\dot{X}} + B_{\dot{\tau}}\right)t_h \geq$$
$$\left[\alpha - \left(nB_A B_{W2} B_{W1} B_{\dot{X}} + B_{\dot{\tau}}\right)\right]t_h$$

Obviously, for all times $t < t_h$, taking into account the sliding mode controller gain α in (11), it follows from (15) that

$$s_c(t)\dot{s}_c(t) = -\alpha|s_c(t)|\left(1 + \sum_{i=1}^{n} A_i(t) w2_i^2(t)\right) + \quad (19)$$
$$\left(\sum_{i=1}^{n} A_i(t) w2_i(t) \sum_{j=1}^{p} w1_{i,j}(t)\dot{x}_j(t) + \dot{\tau}(t)\right)s_c(t) \leq$$
$$\left(-\alpha + nB_A B_{W2} B_{W1} B_{\dot{X}} + B_{\dot{\tau}}\right)|s_c(t)| < 0$$

and a sliding motion exists on $s_c(t) = 0$ for $t > t_h$.

The relation between the sliding line s_p and the zero adaptive learning error level s_c, if λ is taken as $\lambda = \frac{k_P}{k_D}$, is determined by the following equation:

$$s_c = \tau^c = k_D \dot{e} + k_P e = k_D\left(\dot{e} + \frac{k_P}{k_D} e\right) = k_D s_p \quad (20)$$

The tracking performance of the feedback control system can be analyzed by introducing the following Lyapunov function candidate:

$$V_p = \frac{1}{2} s_p^2 \quad (21)$$

Theorem 2.3: If the adaptation strategy for the adjustable parameters of the NNFC is chosen as in equation (10) then the negative definiteness of the time derivative of the Lyapunov function in (21) is ensured.

Proof 2.3: Evaluating the time derivative of the Lyapunov function in (21) yields

$$\dot{V}_p = \dot{s}_p s_p = \frac{1}{k_D^2}\dot{s}_c s_c \leq \quad (22)$$
$$\frac{1}{k_D^2}|s_c|\left(-a + nB_A B_{W2} B_{W1} B_{\dot{X}} + B_{\dot{\tau}}\right) < 0, \forall s_c, s_p \neq 0$$

The obtained result means that, assuming the sliding mode control task is achievable, using τ^c as a learning error for the NNFC together with the adaptation law of (10) enforces the desired reaching mode followed by a sliding regime for the system under control. It is straightforward to prove that the hitting occurs in a finite time (see the second part of proof 2.2).

III. APPLICATION TO NEURO-ADAPTIVE CONTROL OF TWO-LINK MANIPULATOR

Due to the highly nonlinear and coupled dynamics of robot manipulators and the often unknown inertial properties of the objects being manipulated, accurate trajectory tracking is difficult to obtain. A solution to the robot control problem requires combining conventional approaches with new learning techniques in order to achieve good performance.

Connectionist methods with distributed processing provide the implementation tools for modeling the complex input/output relations of the manipulator dynamics. The connectionist approach may, in principle solve the problem of variable - coupling complexity and state - dependency because neural networks through the process of training can approximate input-output mappings. In this way connectionist structure as part of a decentralized feedback control law can compensate wide range of robot uncertainties.

In this section, the effectiveness of the proposed approach is evaluated on the example of neuro-adaptive control of a simple two-link planar robot manipulator shown on Fig. 3. The manipulator was modeled as two rigid links of length 0.5m each with point masses $m_1 = 10$ kg and $m_2 = 8$ kg at their distal ends. The dynamic equations of the manipulator can be found in [16].

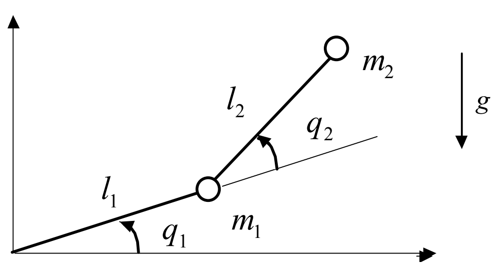


Fig. 3. A two-link manipulator

Two identical neural network structures (one per manipulator joint) with one hidden layer each, consisting of 5 neurons, where implemented as NNFC.

To alleviate the "chattering" phenomena, present in the networks outputs the following standard substitution is adopted for the ideal switch function.

$$sign\,(s_c) \approx \frac{s_c}{|s_c| + \delta} \tag{23}$$

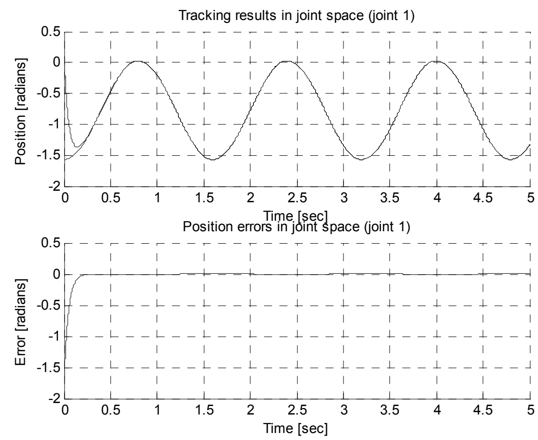
with $\delta = 0.05$.

The value used for the variable structure gain α was chosen $\alpha = 25$ and the sampling time was set to 0.001 s.

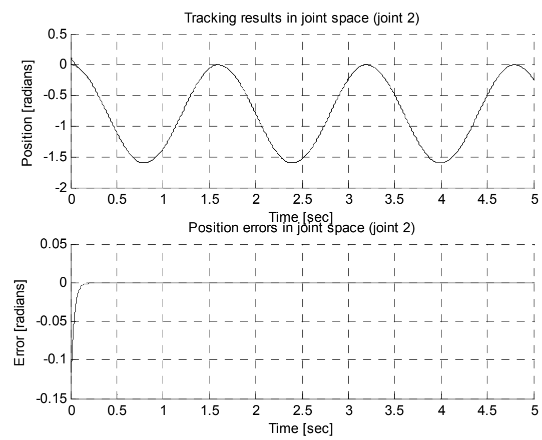
The results are presented on Fig. 4a, b, c where the following denotations are used: Solid lines trajectories are the actual trajectories. Desired trajectories are plotted with dashed lines.

The desired joint position trajectories were chosen as:

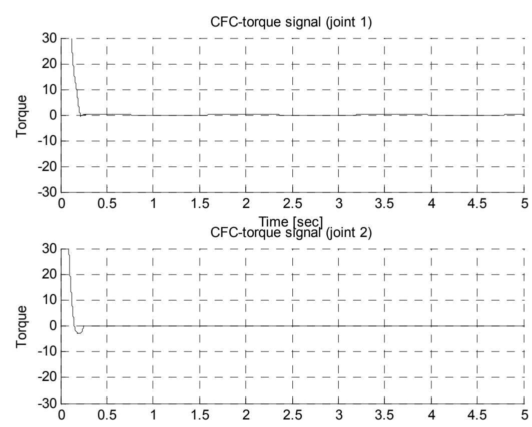
$$\begin{aligned} q_{d1} &= -0.77 + 0.8\sin((2\pi t/1.6) - \pi/2); \\ q_{d2} &= -0.8 - 0.8\sin((2\pi t/1.6) - \pi/2) \end{aligned} \tag{24}$$



a)



b)



c)

Fig. 4. Tracking performance of the proposed sliding mode feedback-error-learning control scheme.

It can be seen that after an initial transition period of approximately 0.25s the CFC torque signals are eliminated and the joint outputs closely follow the required trajectories demonstrating a good tracking performance of the control scheme.

In Fig. 5, the phase space behavior is demonstrated. As it can be seen from the plot the $s_p = 0$ line is the attracting invariant. Clearly the error vector is guided towards the sliding manifold and due to the design presented it is forced to remain in the vicinity of the attracting loci without explicitly knowing the analytical details of the manipulator dynamics.

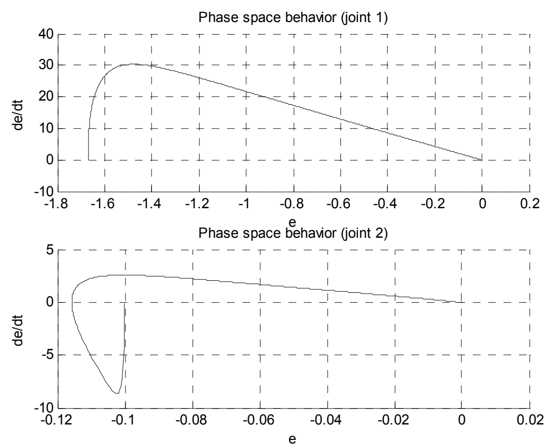


Fig. 5. Phase space behavior

IV. Conclusion

A new learning scheme using sliding mode feedback-error-learning strategy for a NN applied to adaptive nonlinear feedback control has been proposed. The control architecture developed has been simulated and its effect on the trajectory tracking performance of a simple robot manipulator has been evaluated. After the neural network has compensated for the nonlinearity of the controlled object through learning, the responses of the controlled object follow the desired responses supplied by inverse reference model implemented in the conventional feedback controller. The learning scheme does not require perfect preliminary knowledge of the nonlinearity of the system under control. In contrast, it may be noted that the popular model-based approaches for robot control (such as the computed torque technique [16]) can result in poor control performance if the specific model structure selected (on which the controller design is based) does not completely reflect all of the existing dynamics.

V. Acknowledgements

The authors would like to acknowledge Bogazici University Research Fund Project No: 03M109.

References

[1] F. L. Lewis, S. Jagannathan and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*, Taylor & Francis, London, 1999.

[2] J. Y. Hung, W. Gao and J. C. Hung, “Variable structure control: A survey,” *IEEE Trans. on Industrial Electronics*, vol. 40 no. 1, pp. 2-22, 1993.

[3] M. O. Efe and O. Kaynak, “Stabilizing and robustifying the learning mechanisms of artificial neural networks in control engineering applications,” *Int. J. Intelligent Systems*, vol. 15, no. 5, pp. 365-388, 2000.

[4] H. Sira-Ramirez and E. Colina-Morles, “A sliding mode strategy for adaptive learning in Adalines,” *IEEE Trans. on Circuits and Systems – I: Fundamental Theory and Applications*, vol. 42, no. 12, pp. 1001-1012, 1995.

[5] G. G. Parma, B. R. Menezes, and A. P. Braga, “Sliding mode algorithm for training multilayer artificial neural networks,” *Electronics Letters*, vol. 34, no. 1, pp.97-98, 1998.

[6] G. G. Parma, B. R. Menezes, and A. P. Braga, “Neural networks learning with sliding mode control: the sliding mode backpropagation algorithm,” *Int. Journal of Neural Systems*, vol. 9, no. 3, pp. 187-194, 1999.

[7] M. O. Efe, O. Kaynak, and X. Yu, “Sliding mode control of three degrees of freedom anthropoid robot by driving the controller parameters to an equivalent regime,” *ASME J. of Dynamic Systems, Measurement and Control*, vol. 122, pp. 632-640, 2000.

[8] A. V. Topalov and O. Kaynak, “On-line learning in adaptive neurocontrol schemes with a sliding mode algorithm,” *IEEE Trans. on Syst., Man and Cyber.*, (part B), vol. 31, no. 3, pp. 445-450, June 2001.

[9] A. V. Topalov, O. Kaynak, and N. G. Shakev, “Neural network identification, predictive modeling and control with a sliding mode learning mechanism: an application to the robotic manipulators,” in *Proc. First Int. IEEE Symp. on Intelligent Systems*, vol. 1, pp. 102-107, Varna, 2002.

[10] M. I. Jordan “Supervised learning and systems with excess degree of freedom,” (COINS Tech. Rep. no. 88-27), University of Massachusetts at Amherst.

[11] J. S. Albus, “A new approach to manipulator control: The cerebellar model articulation controller,” *Trans. of The American Society of Mech. Eng.*, vol. 97, pp. 220-227, 1975.

[12] W. T. Miller, III, R. P. Hewes, F. H. Glanz and L. G. Kraft, “Real-time dynamic control of an industrial manipulator using a neural-network-based learning controller,” *IEEE Trans. on Robotics & Automation*, vol. 6, no. 1, pp. 1-9, 1990.

[13] A. G. Barto, R. S. Sutton and C. W. Anderson, “Neuronlike adaptive elements that can solve difficult learning control problems,” *IEEE Trans. on Syst., Man and Cybern.*, vol. 13, pp.834-846, 1983.

[14] M. Kawato, Y. Uno, M. Isobe, R. A. Suzuki, “Hierarchical model for voluntary movement and with application to robotics,” *IEEE Control System Magazine*, vol. 8, no. 2, pp. 8-16, 1988

[15] H. Gomi and M. Kawato, “Neural network control for a closed-loop system using feedback-error-learning”, *Neural Networks*, vol. 6, pp. 933-946, 1993.

[16] F. L. Lewis, C. T. Abdallah, and D. M. Dawson, *Control of Robot Manipulators*. Macmillan Publishing Company, 1993.