INCREASING THE NOISE REJECTION CAPABILITY OF NEURAL NETWORKS IN IDENTIFICATION OF NONLINEAR SYSTEMS

Mehmet Onder Efe
Department of Electrical and Electronic Enineering
Bogazici University
Istanbul, 80815, Turkey

Okyay Kaynak
Department of Electrical and Electronic Enineering
Bogazici University
Istanbul, 80815, Turkey

Imre J. Rudas
Dept. of. Info. Technology
Banki Donat Polytechnic
H-1081, Nepszinhaz, u.8
Budapest, Hungary

ABSTRACT

This paper concentrates on the estimation of the state vector of a direct drive robotic manipulator having two degrees of freedom and four state variables. The method presented in the paper uses the Variable Structure Systems methodology in stabilizing the gradient descent based training dynamics of the identifier. Simulations have been carried out with the worst conditions, namely a considerable amount of observation noise and varying payload conditions. The results observed clearly recommend the use of the algorithm presented.

1. INTRODUCTION

During the last few decades, widespread innovations in the realm of *Computational Intelligence* have been witnessed, the approaches of which offer a practical framework for solving complex problems through the use of human expertise and a priori knowledge about the problem in hand. *Artificial Neural Networks* are one of the most popular members of the computational intelligence area because of their ability to represent mappings and internal relations between the data in hand with a number of processing elements, which are called *neurons*. Numerical data in the form of [Input State] ⇒ [Output] statements are learned in the massively interconnected structure of a neural network. Earlier works on the mapping properties of these architectures have shown that neural networks are universal approximators [1-2]. Various architectures of neural systems are studied in the literature. Feedforward and recurrent neural networks [3-4], Gaussian radial basis function neural networks [3-4], Runge-Kutta neural networks [5], and dynamical neural networks [6] constitute typical structurally different models. These models have successfully been applied to problems extending from control applications to image/pattern recognition problems.

In the engineering practice, stability and robustness are of crucial importance. Because of this, the implementation-oriented engineering expert is always in pursuit of a design, which provide accuracy as well as insensitivity to environmental disturbances and structural uncertainties. A suitable way of tackling with uncertainties without the use of complicated models is to introduce Variable Structure Systems (VSS) theory based components into the system structure. The theory has first been introduced by Emelyanov [7] and numerous contributions to VSS theory have been made during the last decade. Some of them are as follows: Hung *et al* [8] have reviewed the control strategy for linear and nonlinear systems. In that study, the switching schemes, putting the differential equations into canonical forms and generating simple VSS based controls are considered in detail. Application of the Sliding Mode Control (SMC) scheme to robotic manipulators and discussion on the quality of the scheme are presented in another work [9]. One of the crucial points in SMC is the selection of the parameters of the sliding surface. Some studies devoted to the adaptive design of sliding surfaces have shown that the performance of control system can be refined by interfacing it with an adaptation mechanism, which regularly redesigns the sliding surface [10-11]. This eventually results in a robust control system. The performance of SMC scheme is proven to be satisfactory in the face of external disturbances and uncertainties in the system model representation. Another systematic examination of SMC approach is presented in [12]. In this reference, the practical aspects of SMC design are assessed for both continuous time and discrete time cases and a special consideration is given to the finite switching frequency, limited bandwidth actuators and parasitic dynamics.

The objective of this paper is to develop a stable training procedure for artificial neural networks, which will enforce the adjustable parameters to settle down to a steady state solution while minimizing an cost function defined by the realization error. This is achieved by performing an appropriate mixture of gradient based parametric displacements [13] and VSS based stabilizing parametric displacements. The early applications of VSS theory in training of computationally intelligent systems have considered the adjustment of the parameters of simple models like Adaptive Linear Elements (ADALINE) [14].

The method presented in [14] also presents the forward and inverse dynamics identification of a Kapitsa pendulum. The fundamental difference of the algorithm presented in this paper is the fact that, the derivation is based on the mixture of two different update values. Furthermore, the eventual form of the parameter update formula alleviates the handicaps of the gradient based training algorithms, which are widely used in most applications in the literature.

This paper is organized as follows: The second section summarizes the conventional method followed in gradient based optimization technique and discusses the typical drawbacks of the approach. The third section presents the derivation of VSS based parameter stabilizing law. In the fourth section, neural network architecture is considered and the relevant formulation for the approach is given. Next section is devoted to the plant to be identified in this study. This is followed by the discussion of simulation studies. Conclusions constitute the last part of the paper.

2. AN OVERVIEW OF GRADIENT DESCENT

In this section, a widely used technique of parameter adjustment, which is called Error Backpropagation (EBP), is briefly reviewed. The method has first been formulated by Rumelhart *et al* [13] in 1980s. The approach has successfully been applied to a wide variety of optimization problems. The algorithm can briefly be stated as follows.

$$e = d - F(\phi, u) \tag{1}$$

As is clearly seen, the error in (1) is the discrepancy between the target signal d and the response of the neural network F. The output F is a continuous function of the network parameters denoted generically by ϕ and inputs denoted by u. The approach defines the cost of realization and the rule of learning as below;

$$J = \frac{1}{2}e^2 \tag{2}$$

$$\Delta\phi = -\eta\frac{\partial J}{\partial \phi} \tag{3}$$

The observation error in (1) is used to minimize the cost function in (2) by utilizing the rule described by (3).

$$\Delta\phi = \eta e \frac{\partial F(\phi, u)}{\partial \phi} \tag{4}$$

The minimization proceeds iteratively as given in (4), for which the sensitivity derivative with respect to the generic parameter ϕ is needed. It is apparent that the method is applicable to the architectures in which the outputs are differentiable with respect to the subject of optimization.

An admissible generalization or learning is achieved in the regions of parameter space where the partial derivative in (4) is smoothly changing. If the system under investigation is a time varying one and if there exists strong environmental disturbances, minimizing the cost in (2) by tuning the adjustable parameters becomes a challenge. An appropriate solution to this problem could be to filter the noisy signal, then to identify the dynamics by using filtered observations. However, the adverse effect of this operation would be the occurrence of a time delay between the desired signal and the neural network response. Since the generalization property can be incorporated into the identification scheme, one would directly seek for alternative training methodologies leading to the reduction of the negative effects of noisy data and nonlinear behavior. In the next section, a method for improving the learning performance for artificial neural networks is presented.

3. VSS FOR PARAMETRIC STABILITY

A continuous-time dynamic model of the parameter update rule prescribed by the gradient descent technique can be written as in (5).

$$\dot{\Delta\phi} = -\frac{1}{T_s}\Delta\phi + \frac{\eta_\phi}{T_s}N_\phi \tag{5}$$

The above model is composed of the sampling time denoted by T_s, the gradient based non-scaled parameter change denoted by $N_\phi = e\partial F(\phi,u)/\partial\phi$ and a scaling factor denoted by η_ϕ, for the selection of which, a detailed analysis is presented in the subsequent discussion. Using Euler's first order approximation for the derivative term, one obtains the following relation, which obviously validates the constructed model in (5) and which leads to the representation in (7).

$$\frac{\Delta\phi(k+1) - \Delta\phi(k)}{T_s} = -\frac{\Delta\phi(k)}{T_s} + \frac{1}{T_s}\eta_\phi N_\phi(k) \tag{6}$$

$$\Delta\phi(k+1) = \eta_\phi N_\phi(k) \tag{7}$$

By comparing (4) and (7), the equivalency between the continuous and discrete forms of the update dynamics is thus clarified. The derivation of the parameter stabilizing component is based on the integration of the system in (5) with variable structure systems methodology. In the design of variable structure controllers, one method that can be followed is the reaching law approach [8]. For the use of this theory in the stabilization of the training dynamics, let us define the switching function as in (8) and its dynamics as in (9).

$$s_\phi = \Delta\phi \tag{8}$$

$$\dot{s}_\phi = -\frac{Q_\phi}{T_s}\tanh\left(\frac{s_\phi}{\varepsilon}\right) - \frac{K_\phi}{T_s}s_\phi = \dot{\Delta\phi} \tag{9}$$

In above, Q_ϕ and K_ϕ are the gains, and ε is the width of the boundary layer. Equating (9) and (5) and solving for $\Delta\phi$ yields the following;

$$\Delta\phi = \eta_\phi N_\phi + Q_\phi \tanh\left(\frac{s_\phi}{\varepsilon}\right) + K_\phi s_\phi \tag{10}$$

With the equality given in (10), the update dynamics is forced to behave as that defined by (9), which is actually a stable dynamics defined by the adopted switching function. In the derivations presented below, a key point is the fact that the system described by (5) is also driven by η_ϕ, which is known as learning rate in the related literature. Now we demonstrate that some special selection of this quantity leads to a rule that minimizes the magnitude of the parametric displacement. Let us define the following quantity for keeping analytic comprehensibility;

$$A_\phi = Q_\phi \tanh\left(\frac{\Delta\phi}{\varepsilon}\right) + K_\phi \Delta\phi \tag{11}$$

Now we have a model described by (5), and an equality formulated by (10). If one chooses a positive definite Lyapunov function as given by (12), the time derivative of this function must be negative definite for stability of parameter change ($\Delta\phi$) dynamics. Clearly the stability in parameter change space implies the convergence in system parameters.

$$V_\phi = \frac{1}{2} s_\phi^2 = \frac{1}{2}(\Delta\phi)^2 \tag{12}$$

$$\dot{V}_\phi = (\Delta\phi)(\dot{\Delta\phi}) \tag{13}$$

If (5) and (10) are substituted into (13), the constraint stated in (14) is obtained for stability in the Lyapunov sense.

$$\eta_\phi^2 + \frac{1}{N_\phi}\left(A_\phi - \Delta\phi\right)\eta_\phi - \frac{1}{N_\phi^2} A_\phi \Delta\phi < 0 \tag{14}$$

Equation (14) can be rewritten in a more tractable form as follows.

$$\left(\eta_\phi + \frac{1}{N_\phi} A_\phi\right)\left(\eta_\phi - \frac{1}{N_\phi}\Delta\phi\right) < 0 \tag{15}$$

Since A_ϕ and $\Delta\phi$ have the same signs, the roots of the expression (15) clearly have opposite signs. The expression on the left-hand side assumes negative values between the roots. Therefore, in order to satisfy the inequality in (15), the learning rate must satisfy the constraint given in (16).

$$0 < \eta_\phi < \min\left\{\left|\frac{1}{N_\phi}\Delta\phi\right|, \left|-\frac{1}{N_\phi}A_\phi\right|\right\} \tag{16}$$

In (16), the interval of learning rate is restricted to positive values. This is due to preserve the compatibility between the gradient based approaches and the proposed approach. An appropriate selection of η_ϕ could be as follows:

$$\eta_\phi = \beta \min\left\{\left|\frac{1}{N_\phi}\Delta\phi\right|, \left|-\frac{1}{N_\phi}A_\phi\right|\right\}, \quad 0 < \beta < 1 \tag{17}$$

By substituting the learning rate formulated in (17) into the stabilizing solution given in (10), the stabilizing component $\Delta\phi_{VSS}$ of the parameter change formula is obtained as;

$$\Delta\phi_{VSS} = \beta \min\left(|\Delta\phi|, |A_\phi|\right)\operatorname{sgn}\left(N_\phi\right) + A_\phi \tag{18}$$

where, $\Delta\phi$ on the right-hand side is the final update value yet to be obtained. The law introduced in (18) minimizes the cost of stability, which is the Lyapunov function defined by (12). The question now reduces to the following; can this law minimize the cost defined by (2)? The answer is obviously not, because the stabilizing criteria in (18) is derived from the displacement of the parameter vector denoted by $\Delta\phi$, whereas the minimization of (2) is achieved when ϕ tends to ϕ^* regardless of what the displacement is. In order to minimize (2), the parameter change anticipated by gradient based optimization technique, which is reviewed in the second section, should somehow be integrated into the final form of parameter update mechanism. As introduced in the second section, error backpropagation algorithm (EBP) evaluates a parameter change as given in (20).

$$\Delta\phi_{EBP} = \zeta N_\phi \tag{19}$$

where, ζ is the constant learning rate in the conventional sense. Combining the laws formulated in (18) and (19) in a weighted average, the eventual parameter update law in (20) is obtained.

$$\Delta\phi = \frac{\alpha_1 \Delta\phi_{VSS} + \alpha_2 \Delta\phi_{EBP}}{\alpha_1 + \alpha_2} \tag{20}$$

The parameter update formula given by (20) carries mixed information containing both the parametric convergence, which is introduced by VSS part, and the cost minimization, which is due to the error backpropagation technique. The balancing in this mixture is left to the designer by an appropriate selection of α_1 and α_2.

4. TRAINING OF FEEDFORWARD NEURAL NETWORKS

In this section, a multilayer perceptron is introduced as the intelligent identifier, the parameters of which are to be updated by using the technique presented. In [15], Narendra and Parthasarathy demonstrate that this structure can effectively be used for identification and control purposes. In the conventional error backpropagation technique, propagating the output error back through the neural network minimizes the cost function given in (2). Based on the derivation presented in detail in [13], the delta values for the output and hidden layer neurons are evaluated as given by (21) and (22) respectively.

$$\delta_j^{k+1,p} = (d_j^p - F_j^p)\,\Psi'(S_j^{k+1,p}) \tag{21}$$

$$\delta_j^{k+1,p} = \left(\sum_{h=1}^{\#neurons_{k+2}} \delta_h^{k+2,p} w_{jh}^{k+1}\right)\Psi'(S_j^{k+1,p}) \tag{22}$$

Having evaluated the delta values during the backward pass, the weight update rule described by (23) is applied for each training pair.

$$\Delta w_{ij\,EBP}^{k} = \zeta\, \delta_j^{k+1,p}\; o_i^{k,p} \tag{23}$$

The eventual form of the update formula can now be constructed by using the equations (18) through (20).

5. NONLINEAR SYSTEM MODEL

In this study, a two degrees of freedom direct drive SCARA robotic manipulator is used as the test bed. Since the dynamics of such a mechatronic system is modeled by nonlinear and coupled differential equations, identifying the dynamics becomes a difficult objective due to the strong interdependency between the variables involved, and the existence of abruptly changing payload conditions and noisy observations. Besides, the ambiguities on the friction related dynamics in the plant model make the design much more complicated. Therefore the methodology adopted must be intelligent in some sense.

The general form of robot dynamics is described by (24) where $M(\theta)$, $V(\theta,\dot{\theta})$, $\tau(t)$ and f stand for the state varying inertia matrix, vector of coriolis terms, applied torque inputs and friction terms respectively.

$$M(\theta)\ddot{\theta} + V(\theta,\dot{\theta}) = \tau - f \tag{24}$$

If the angular positions and angular velocities are described as the state variables of the system, four coupled and first order differential equations can define the model. In (25) and (26), the terms seen in (24) are given explicitly.

$$M(\theta) = \begin{bmatrix} p_1 + 2p_3 cos(\theta_2) & p_2 + p_3 cos(\theta_2) \\ p_2 + p_3 cos(\theta_2) & p_2 \end{bmatrix} \tag{25}$$

$$V(\theta,\dot{\theta}) = \begin{bmatrix} -\dot{\theta}_2\left(2\dot{\theta}_1 + \dot{\theta}_2\right)p_3 \sin(\theta_2) \\ \dot{\theta}_1^2 p_3 \sin(\theta_2) \end{bmatrix} \tag{26}$$

In above, p_1 = 2.0857 + 0.0576 M_p, p_2 = 0.1168 + 0.0576 M_p and p_3 = 0.1630 + 0.0862 M_p. Here M_p denotes the mass of the payload. The values of which are illustrated in Fig. 1. The details of the plant model are presented in [16].

6. SIMULATION STUDIES

In the simulations, the dynamics of the system introduced in Sec. 5 is identified by the neural network model discussed in Sec. 4. During the simulations, the main objective is to keep the update dynamics in a stable region together with the clarification of the noise rejection capability of the update mechanism presented. This is achieved through a suitable combination of conventional gradient based learning strategy with that based on the variable structure systems methodology.

In the simulations performed the system under investigation has been kept under an external control loop while the identifier is extracting the dynamical behavior. The state vector and the torque inputs are applied to the neural network. In response to this input, the identifier tries to match the next state vector to which is actually responded by the manipulator. During this process, the state vector is corrupted by a random noise, whose maximum value is restricted to the ±10% of the observed state value. The noise sequence has zero mean and has a uniformly distributed characteristic. Initially, the neural network weights and biases are set to small numbers. Next, the on-line training phase takes place. During this phase, the parameters of the network are adjusted such that the identifier best realizes the actual value of the state vector. For this purpose, a neural network, which has 6 inputs, 12 hidden neurons with hyperbolic tangent neuronal nonlinearities and 4 linear output neurons is employed.

The reference velocity trajectory is a trapezoidal one, and in all simulations the system is started with zero initial state errors.

In Fig. 2 the response of the manipulator under control is illustrated. In Fig. 3, the results observed with the use of only the gradient descent are depicted. If the magnitudes in Figs. 2 and 3 are considered, one can directly infer that the ordinary gradient descent algorithm fails under the conditions stated above. Fig. 4 depicts the realization errors for the mixed training algorithm. The improvement introduced by the method is clear from the magnitudes of the error profiles. The settings of both cases are tabulated in Table 2, where in the former α_I=0, and in the latter α_I=20.

During the on-line training of the identifier, the squared sum of parametric changes is defined to be the cost of stability. The cost function is described by (27), in which the summation is over all adjustable parameters of the neural network. The time behavior of the cost of stability is illustrated in Figs. 5 and Fig. 6 for α_I=0 and α_I=20 cases respectively.

$$J_s(t) = \sum_{\phi} \left(\Delta\phi(t)\right)^2 \tag{27}$$

As can be deduced from Figs. 5 and 6, the parametric stabilization performance of the proposed methodology is highly promising, which means that the introduced approach achieves the subject of optimization with less effort.

7. CONCLUSIONS

In this paper, a novel technique for improving the learning performance of artificial neural networks is presented. An approximate model of ordinary gradient based training procedure (EBP) is constructed and variable structure systems approach is incorporated into the proposed form of the parameter update law. In this procedure, error backpropagation rule is responsible for the minimization of squared realization error while the variable structure systems based law is responsible for the stability in the parameter space.

The conventional approaches suffer from some handicaps, such as imperfect modeling, noisy observations or time varying parameters. If the effects of these factors are transformed to the cost hypersurface, whose dimensionality is determined by the adjustable design parameters, it is evident that the surface may have directions along which the sensitivity derivatives can assume large values. In these cases, gradient based approaches evaluate large parametric displacements, which can eventually lead to a locally divergent behavior requiring an excessive tuning effort. In control engineering practice, such a behavior constitutes a potential danger from a safety point of view. The approach presented in this paper takes care of the instantaneous fluctuations in the parameter space. Since the variable structure systems approach is well known with its robustness property, an appropriate combination of gradient rule and variable structure systems can eliminate the handicaps stated above. The fluctuations that are most likely to occur in the parameter space during training are dampened out. The combination is therefore a good candidate for efficient parameter tuning.

In the application example presented, feedforward neural network structure is utilized as the computationally intelligent architecture. It must be emphasized that the task is accomplished with 12 hidden neurons in the FNN structure and the tuning of the parameters is performed on-line. The results presented confirm the prominent features of the approach, such as, high noise rejection capability, generalization of a complex system dynamics by input-output data, and capability of tolerating the adverse effects of varying payload mass. The algorithm is applicable to any neural network model provided that the model output is differentiable with respect to the parameter of interest.

8. REFERENCES

[1] Hornik, K., “Multilayer Feedforward Networks are Universal Approximators,” *Neural Networks*, v.2, pp 359-366, 1989.

[2] Funahashi, K., “On the Approximate Realization of Continuous Mappings by Neural Networks,” *Neural Networks*, v.2, pp 183-192, 1989.

[3] Haykin, S., Neural Networks, Macmillan College Printing Company, New Jersey, 1994.

[4] Efe, M. O., and O. Kaynak, “A Comparative Study of Neural Network Structures in Identification of Nonlinear Systems,” *Mechatronics*, v.9, no.3, pp.287-300, 1999.

[5] Wang, Y-J. and C-T. Lin “Runge-Kutta Neural Network for Identification of Dynamical Systems in High Accuracy,” *IEEE Transactions on Neural Networks*, v.9., no. 2, pp. 294-307, March 1998.

[6] Gupta, M. M., D. H. Rao, “Dynamical Neural Units with Applications to the Control of Unknown Nonlinear Systems,” *Journal of Intelligent and Fuzzy Systems*, v.1, no.1, pp. 73-92, 1993.

[7] Emelyanov, S. V., *Variable Structure Control Systems*, Moscow, Nauka, 1967.

[8] Hung, J. Y., W. Gao, J. C. Hung, “Variable Structure Control: A Survey,” *IEEE Trans. on Industrial Electronics*, v.40, no. 1, pp.2-22, Feb. 1993.

[9] Gao, W., J. C. Hung, “Variable Structure Control of Nonlinear Systems: A New Approach,” *IEEE Trans. on Industrial Electronics*, v.40, no. 1, pp.45-55, Feb. 1993.

[10] Kaynak, O., F. Harashima and H. Hashimoto, “Variable Structure Systems Theory, as Applied to Sub-time Optimal Position Control with an Invariant Trajectory,” *Trans. IEE of Japan*, Sec. E, v.104, no.3/4, pp.47-52, 1984.

[11] Bekiroglu, N., “Adaptive Sliding Surface Design for Sliding Mode Control Systems,” Ph.D. Thesis, Bogazici University, 1996.

[12] Young, K.D., V. I. Utkin and U. Ozguner, “A Control Engineer’s Guide to Sliding Mode Control,” *IEEE Trans. on Control Systems Technology*, v.7, no.3, pp.328-342, May 1999.

[13] Rumelhart, D. E., G. E. Hinton and R. J. Williams, “Learning Internal Representations by Error Propagation,” in D. E. Rumelhart and J. L. McClelland, eds., *Parallel Distributed Processing*, v.1, pp.318-362, MIT Press, Cambridge, M.A., 1986.

[14] Sira-Ramirez, H. and E. Colina-Morles, “A Sliding Mode Strategy for Adaptive Learning in Adalines,” *IEEE Trans. on Circuits and Systems – I: Fundamental Theory and Applications*, v.42, no.12, pp.1001-1012, Dec. 1995

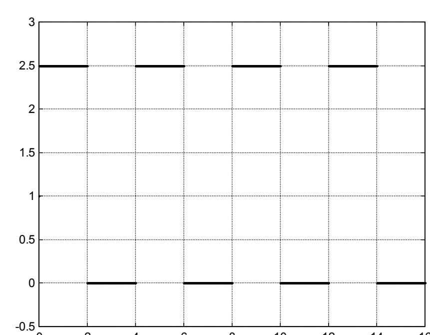
[15] Narendra, K. S., K. Parthasarathy, “Identification and Control of Dynamical Systems Using Neural Networks,” *IEEE Trans. on Neural Networks*, v.1, no. 1, pp.4-27, Mar. 1990.

[16] Direct Drive Manipulator R&D Package User Guide, Integrated Motions Incorporated, 704 Gillman Street, Berkeley, California 94710, U.S.A.

8. ACKNOWLEDGMENTS

This work is supported in part by Bogazici University Research Fund (Project No: 99A202) and a grant from TUBITAK, Turkey and OMFB, Hungary.

Payload Mass (M_p) in Time



Time (sec)

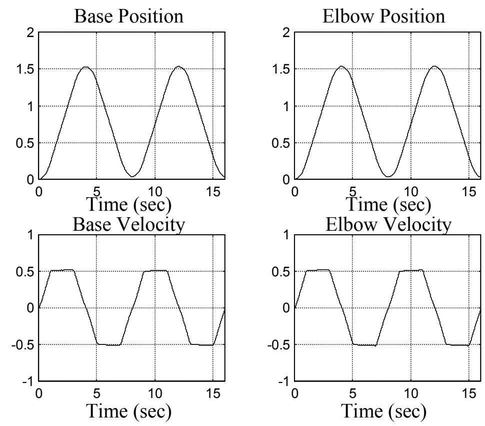


Figure 2. Response of the Manipulator

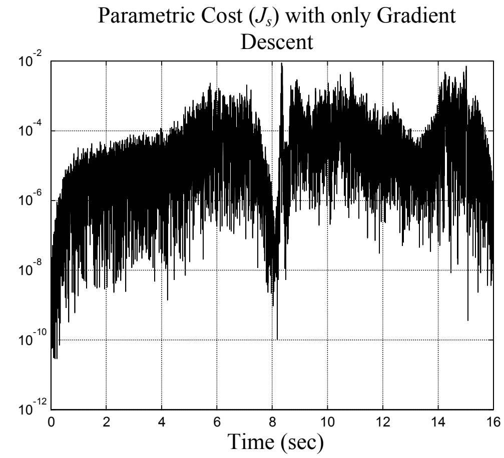


Figure 5. Cost of Stability with only Gradient Descent

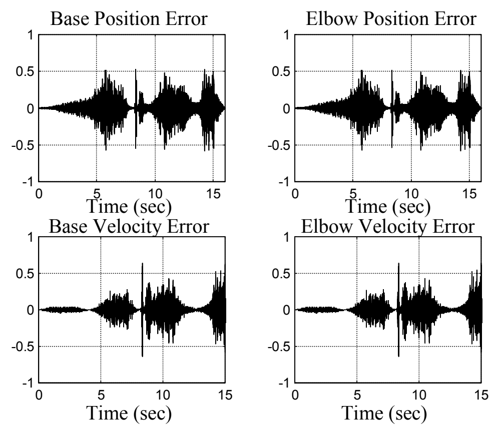


Figure 3. Discrepancy Between the Response of the Manipulator and the Identifier Outputs with only Gradient Descent Based Tuning

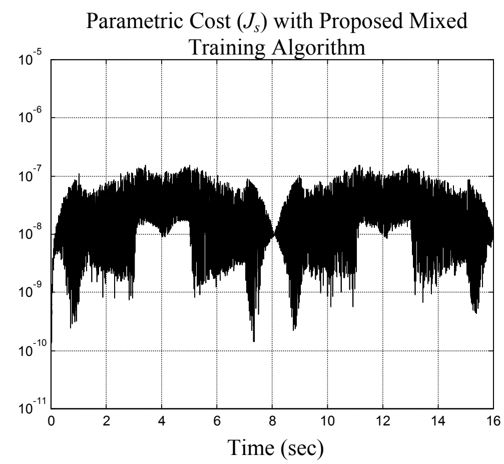


Figure 6. Cost of Stability with Mixed Training Strategy

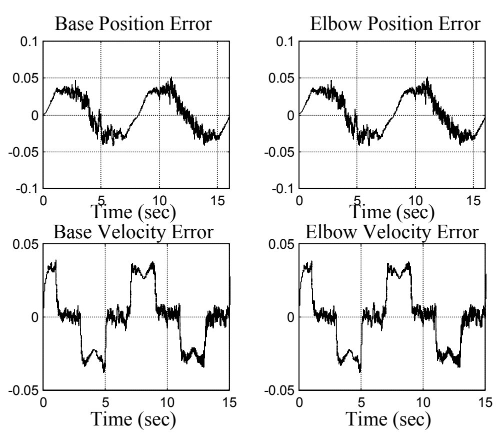


Figure 3. Discrepancy Between the Response of the Manipulator and the Identifier Outputs with Proposed Mixed Tuning Algorithm