Adaptive Backstepping Controller of Discrete-Time Nonlinear Systems with Input Saturation

(IJCSNT) International Journal of Communication Systems and Network Technologies
ISSN Number: 2053-6283
Volume 1, Issue No. 2, 87–100

https://journals.mirlabs.in/index.php/ijcsnt
DOI: 10.18486/ijcsnt/1.2.008
IJCSNT

Vinay Kumar Deolia[1], Shubhi Purwar[2], T. N. Sharma[3]

Abstract

This paper proposes a backstepping controller for the class of discrete-time nonlinear system in the presence of saturation constraints. In this paper, hyperbolic tangent function is used to limit the amplitude saturation constraints. The actuator saturation is assumed to be unknown and compensated by a pre compensator using Chebyshev neural network (CNN). The unknown nonlinear functions are also approximated by CNN. Weight update laws, based on Lyapunov theory are derived to make this scheme adaptive and the convergence properties are shown. Simulation results validate the effectiveness of proposed scheme.

Keywords

Backstepping Controller, Chebyshev Neural Network (CNN), Actuator Saturation, Lyapunov Stability.

Received: 25 May 2012; **Revised:** 29 July 2012; **Accepted:** 20 August 2012;
Published: 30 August 2012;

[1]Deptt. of Electronics and Commn. Engg. MNNIT, Allahabad, India.
[2]Deptt. of Electrical Engg. MNNIT, Allahabad, India.
[3]Deptt. of Electronics and Commn. Engg. MNNIT, Allahabad, India.

Corresponding author:
Email: vkdeolia@indiatimes.com

Introduction

It is well recognized that actuator not only degrades the performance of the control system but may also lead to instability of the closed loop system. Several results on the actuator saturation problem for linear systems have been reported in [1–4] and the references therein. On the other hand the actuator saturation problem for nonlinear system has received less attention in literature.

The actuator saturation problem has been tackled for nonlinear continuous time systems in [5–7]. In the global stabilization of nonlinear system with input saturation via state and output feedback is addressed in [5]. In [6] a nonlinear antiwindup scheme is proposed to guarantee stability and local performance recovery on Euler-Lagrange systems subject to input magnitude saturation. Neural network (NN) controller for nonlinear systems in normal form with actuator saturation is presented in [7]. A wavelet based controller for nonlinear systems taking well defined relative degree is proposed in [8].

As far as continuous time systems are concerned, the control laws are generally derived by using Lyapunov stability theory. Therefore the stability of closed-loop system is easily guaranteed for discrete-time systems, adaptive control is more complex due to the noncasual problem. Only a few results have been obtained for actuator saturation in discrete-time nonlinear systems. In [9], state feedback and output feedback dynamic NN control laws are established for pure feedback systems with actuator saturation. Neural network (NN) controller for nonlinear discrete-time systems in normal form with the input constraints is proposed in [10]. Shuzi Sam Ge et al [11], proposed the work for a class of nonlinear discrete-time systems with unknown control directions. Adaptive control employing the predicted future states was implemented by transforming the system to n-step-ahead predictor.

In recent decades, adaptive control of discrete-time system has been studied extensively and lately, nonlinear discrete-time systems in strict-feedback have attracted much research interest. Using NN approximation, controlling strict-feedback system with unknown system functions has been studied in [12]. The result has also been extended to multi input multi output (MIMO) systems in [13]. The Chaos control in atomic force microscope (AFM) system using backstepping technique has been dealt in [14].

In this paper a controller is developed for discrete-time strict-feedback nonlinear system with unknown system nonlinearities as well as input saturation. The unknown nonlinearities are approximate using Chebyshev Neural Network (CNN). The input saturation is tackled using CNN saturation compensator. The basic idea of CNN saturation compensator is to introduce control modifications in order to recover, as much as possible, the performance induced by backstepping design carried out on the basis of the unsaturated system.

This paper is organized as follows. CNN structure and its approximation properties are given in Section 2. The controller design of proposed scheme is presented in Section 3. In Section 4 stability analysis is discussed and simulation results are presented in Section 5. The note ends with conclusion.

CNN Approximation

The structure of CNN is shown in Fig.1. CNN is a functional link network (FLN) based on Chebyshev polynomials. There are two main parts in the architecture of CNN, namely, numerical transformation part and learning part. Numerical transformation deals with input to hidden layer by approximate transformable method. The transformation is a functional expansion (FE) of input pattern comprising of a finite set of Chebyshev polynomials. As a result the Chebyshev polynomials basis can be viewed as a new input vector. The learning part is a functional-link neural network based on Chebyshev polynomials[15–17]. The Chebyshev polynomials can be obtained by a recursive formula

$$T_{i+1}(x) = 2x\,T_i(x) - T_{i-1}(x), \quad T_0(x) = 1 \tag{1}$$

where, $T_i(x)$ is a Chebyshev polynomials, i is the order of polynomials chosen and here is a scalar quantity. The different choices of $T_1(x)$ are x & $2x$.

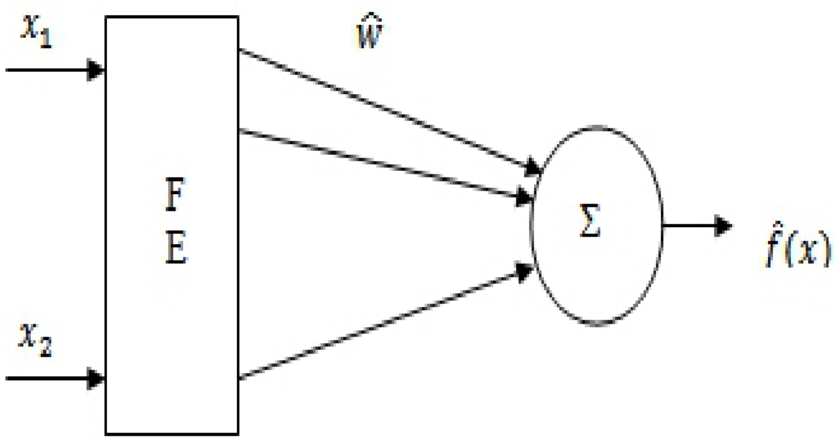

Figure 1. Chebyshev Neural Network

In Fig. 1 the output of single layer neural network is given by

$$\hat{f}(x) = \hat{w}^T \varphi \tag{2}$$

where, $\hat{w}$ are the weights of neural network. Based on the approximation property of CNN[15–17], there exist ideal weights w, so that the function to be approximated can be represented as

$$f(x) = w^T \varphi + \varepsilon \tag{3}$$

where, ε is the CNN functional reconstruction error vector and $\|\varepsilon\| \leq \varepsilon_N$ is bounded.

In CNN functional expansion of the input increases the dimension of input pattern. So with the help of CNN it is very simple to approximate a complex nonlinear systems.

Controller Design

Saturation Nonlinearity:

Fig.2 shows the nonlinear saturation, where $\bar{\Gamma}$ and $\underline{\Gamma}$ are scalars. In general $\bar{\Gamma}$ and $\underline{\Gamma}$ can be vectors. In this paper, saturation nonlinearity is unknown. If the control falls outside the range of the actuator then

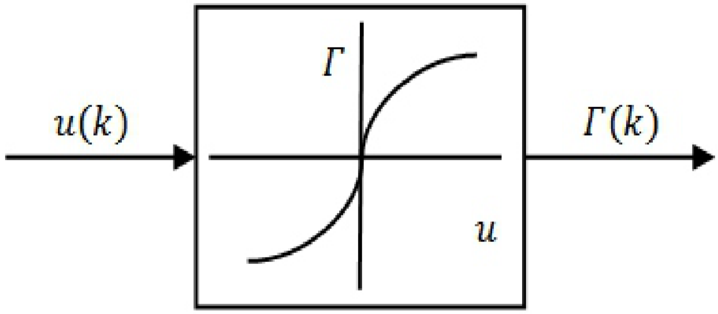

Figure 2. Saturation nonlinearity

actuator saturation occurs. It means that is not completely implemented by the device[7]. The control, which cannot implemented by actuator, represented by $\delta(k)$ is given by

$$\delta(k) = \Gamma(k) - u(k) \tag{4}$$

In this paper CNN is used to approximate the function $\Gamma(k)$.

Mathematically, the output of the actuator $\Gamma(k)$ is given by

$$\Gamma(k) = A * \tanh(u(k)) \tag{5}$$

The control $u(k)$ is subjected to amplitude limitation $|\Gamma(k)| < A$, where $A > 0$ denotes the control amplitude bound.

Nonlinear Systems Dynamics:

Consider the following single-input and single-output (SISO) discrete-time nonlinear system in strict-feedback form[12]:

$$\begin{aligned} x_i(k+1) &= f_i(x_i(k)) + g_i(x_i(k))x_{i+1}(k), \\ & i = 1, 2, \ldots, n-1, \\ x_n(k+1) &= f_n(x_n(k)) + g_n(x_n(k))\Gamma(k), \\ y(k) &= x_1(k), \end{aligned} \tag{6}$$

where $x_i(k) = [x_1(k), x_2(k) \ldots x_i(k)]^T \in R^i$, $i = 1, 2, \ldots, n-1$, are the state variables $\Gamma(k) \in R$ & $y(k) \in R$, are system input and output respectively. $f_{(i)}$ and $g_{(i)}$ $i = 1, 2, \ldots n-1$, are the nonlinear functions that contains both parametric and nonparametric uncertainties. f_i are unknown nonlinear functions and g_i are assumed to be known and invertible.

Equation (6) is known as strict-feedback form because f_i and g_i, depend only on, that is, on state variables that are “fed-back.” The control objective is to make $x_1(k)$ to follow certain desired trajectory $x_{1d}(k)$ is assumed to be bounded and known.

Tracking Error Dynamics:

Define tracking error vector

$$r(k) = [e_1(k)\ e_2(k) \dots e_n(k)]^T \tag{7}$$

$$e_1(k) = x_1(k) - x_{1d}(k)$$
$$e_2(k) = x_2(k) - x_{2d}(k)$$
$$\vdots$$
$$e_i(k) = x_i(k) - x_{id}(k)$$

where $x_{id}(k)$ can be obtained from (16)
Similarly,

$$r(k+1) = [e_1(k+1)\ e_2(k+1) \dots e_n(k+1)]^T \tag{8}$$

$$e_i(k+1) = x_i(k+1) - x_{id}(k+1)$$
$$i = 1, 2, 3, \dots, n-1.$$
$$e_n(k+1) = x_n(k+1) - x_{nd}(k+1)$$

Substituting $x_n(k+1)$ from (6) in (10)

$$e_n(k+1) = f_n(x_n(k)) + g_n(x_n(k))\Gamma(k) - x_{nd}(k+1)$$

Using (4) in the above equation,

$$e_n(k+1) = f_n(x_n(k)) + g_n(x_n(k))(u(k) + \delta(k)) - x_{nd}(k+1) \tag{9}$$

Let $u(k) + \delta(k) = u'(k)$, then tracking control law will be

$$u'(k) = \frac{1}{g_n(x_n(k))}\left[-\hat{f}_n(x_n(k)) + x_{nd}(k+1) - k_{n-1}e_{n-1}(k+1)\right] \tag{10}$$

$x_{nd}(k+1)$ which is the future value of fictitious controller x_{nd}, and the unknown nonlinear functions f_i, $i = 1, 2, \dots n-1$, are approximated using CNN. For some constant "ideal" weights w_i, $i = 1, 2, \dots, n-1$, i.e.,

$$\hat{f}_i = w_i^T \varphi_i + \varepsilon_i, \quad \|\varepsilon_i\| < \varepsilon_{iN} = \text{constant}, \quad i = 1, 2, \dots, n-1 \tag{11}$$

where φ_i's provide suitable basis functions for the NNs.
Assumption1: (Bounded Ideal NN Weights): The ideal NN weights w are bounded so that

$$\|w\|_F \le w_M \tag{12}$$

with W_n a known bound.
The symbol $\|\cdot\|_F$ denotes the Frobenius norm, i.e. given a matrix A, the Frobenius norm is given by[18],

$$\|A\|_F^2 = tr(A^T A) \tag{13}$$

Assumption2: (Bounded Estimation Error): The functional estimation error satisfies $\|\hat{f}_i(x)\| \leq f_M(x)$.

where,

$$\tilde{f}_i(x) = f_i(x) - \hat{f}_i(x) \tag{14}$$

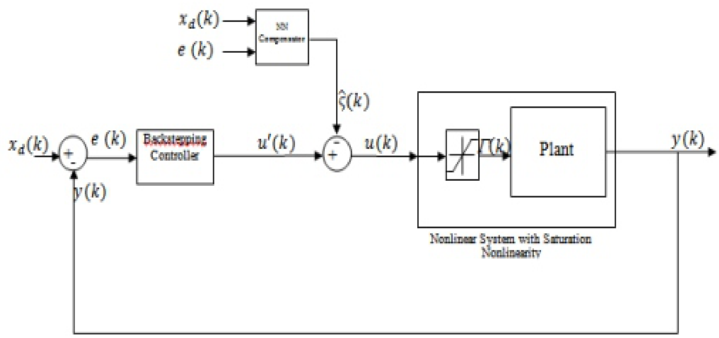

Figure 3. Back stepping NN control of nonlinear system with NN saturation compensator

The net reconstruction errors ε_i are bounded by known constants ε_{iN}, $i = 1, 2, \ldots, n-1$.

Define the NN functional estimate of f_i in (13) by

$$\hat{f}_i = \hat{w}_i \varphi_i, \quad i = 1, 2, \ldots, n-1$$

with $\hat{w}_i$ the current NN weight estimates provide by tuning algorithms.

Here the objective is to design control $\Gamma(k)$ to make the system output $y(k)$ follow a known and bounded trajectory $x_{1d}(k)$. But the causality contradiction is the biggest problem in discrete time domain, when controller is designed for general strict feedback form of nonlinear systems using backstepping[12]. In equation (12) $x_{nd}(k+1)$ is the future value of fictitious control $x_{nd}(k)$.

Therefore it is infeasible in practice to design $u'(k)$. This problem can be avoided by approximate the fictitious controller $x_{nd}(k)$ through CNN. For first $(n-1)$ equations of (6) design an ideal fictitious control of the form,

$$\begin{aligned}
x_{2d}(k) &= \frac{1}{g_1(\bar{x}_1(k))}\left[-\hat{f}_1(\bar{x}_1(k)) + x_{1d}(k+1) - k_1 e_1(k)\right] \\
x_{3d}(k) &= \frac{1}{g_2(\bar{x}_2(k))}\left[-\hat{f}_2(\bar{x}_2(k)) + x_{2d}(k+1) - k_2 e_2(k)\right] \\
&\vdots \\
x_{id}(k) &= \frac{1}{g_{i-1}(\bar{x}_{i-1}(k))}\left[-\hat{f}_{i-1}(\bar{x}_{i-1}(k)) + x_{(i-1)d}(k+1) - k_{i-1} e_{i-1}(k)\right] \\
&\vdots \\
x_{nd}(k) &= \frac{1}{g_{n-1}(\bar{x}_{n-1}(k))}\left[-\hat{f}_{n-1}(\bar{x}_{n-1}(k)) + x_{(n-1)d}(k+1) - k_{n-1} e_{n-1}(k)\right]
\end{aligned} \tag{15}$$

The equation (16) can be write in the form of one-step ahead predictor,

$$x_{nd}(k+1) = \frac{1}{g_{n-1}(x_{n-1}(k+1))}\left[-\hat{f}_{n-1}(\bar{x}_{n-1}(k+1)) + x_{(n-1)d}(k+1) - k_{n-1} e_{n-1}(k+1)\right] \tag{16}$$

where $g_{n-1}(x_{n-1}(k+1))$ is assumed to be known & constant, $f_{n-1}(x_{n-1}(k+1))$ is the approximated using CNN. Design parameter $|k_{n-1}| \leq 1$.

Substituting (17) in (12) gives the output of backstepping controller $u'(k)$ as in shown in Fig.3.

The control signal $u(k)$ is composed of the tracking controller and saturation compensator in Fig.3.

$$u(k) = u'(k) - \hat{\varsigma}(k) \tag{17}$$

where $\hat{\varsigma}(k)$ is the approximated value of modified saturation nonlinear function $\delta(k)$.

Neural Network Saturation Compensator:

Using the CNN function approximation property, there exist a CNN that closely approximates the modified saturation nonlinear function $\delta(k)$.

The inputs are $\bar{x} = [x_{1d}\ e]^T$ and the output can be represent as

$$\delta(k) = w^T \varphi(\bar{x}) + \varepsilon \tag{18}$$

where w is the weight vector, ε is the reconstruction error of neural network. The $\delta(k)$, which is the approximation of is given by

$$\hat{\varsigma}(k) = \hat{w}^T \varphi(\bar{x}) \tag{19}$$

The weight approximation error is

$$\tilde{w} = w - \hat{w} \tag{20}$$

From (16),

$$x_{id}(k+1) = x_{(i+1)d}(k)g_i(x_i(k)) + \hat{f}_i(x_i(k)) + k_i e_i(k)$$
$$\vdots$$
$$x_{nd}(k+1) = g_n(x_n(k))(u(k) + \hat{\varsigma}(k)) + \hat{f}_n(x_n(k)) + k_n e_n(k)$$

Substituting (22), (8), (4) & (6) in (10),

$$\begin{aligned} e_i(k+1) &= \tilde{f}_i(x_i(k)) + g_i(x_i(k))e_{i+1}(k) - k_i e_i(k) \\ &\vdots \\ e_n(k+1) &= \tilde{f}_n(x_n(k)) + g_n(x_n(k))\hat{\varsigma}_1(k) - k_n e_n(k) \end{aligned} \tag{21}$$

where,

$$\hat{\varsigma}_1(k) = \delta(k) - \hat{\varsigma}(k) \tag{22}$$

The functional estimation errors $\tilde{f}_i(x_i(k))$ can be represented by NN for the values of ideal weights $w_i(k)$,

$$\begin{aligned} \tilde{f}_i(x_i(k)) &= \tilde{w}_i^T(k)\varphi_i(x_i(k)) + \varepsilon_i(k) \\ &\vdots \\ \tilde{f}_n(x_n(k)) &= \tilde{w}_n^T(k)\varphi_n(x_n(k)) + \varepsilon_n(k) \end{aligned} \tag{23}$$

Substitute (25) in (23),

$$\begin{aligned} e_i(k+1) &= \tilde{w}_i^T(k)\varphi_i(x_i(k)) + \varepsilon_i(k) + g_i(x_i(k))e_{i+1}(k) - k_i e_i(k) \\ &\vdots \\ e_n(k+1) &= \tilde{w}_n^T(k)\varphi_n(x_n(k)) + \varepsilon_n(k) + g_n(x_n(k))\hat{\varsigma}_1(k) - k_n e_n(k) \end{aligned} \tag{24}$$

The above equation can be written in compact form as

$$r(k+1) = \tilde{w}(k)\varphi(x(k)) + \varepsilon(k) + Hr(k) + P\hat{\varsigma}_1(k) - Kr(k) \tag{25}$$

where, $r(k+1)$ is given in (9).

$$\begin{aligned} \tilde{w}(k) &= diag\{\tilde{w}_1, \tilde{w}_2 \ldots \tilde{w}_n\} \\ \varepsilon(k) &= [\varepsilon_1(k)\ \varepsilon_2(k) \ldots \varepsilon_n(k)]^T \\ \varphi(x(k)) &= [\varphi_1^T(k)\ \varphi_2^T(k) \ldots \varphi_n^T(k)]^T \\ K &= diag\{k_1\ k_2 \ldots k_n\} \end{aligned}$$

H is a matrix defined by

$$H = \begin{bmatrix} 0 & g_1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & g_2 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & 0 & & g_i & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 0 & \cdots & g_n \end{bmatrix}$$

and P is defined by

$$P = \begin{bmatrix} 0 & \cdots & \cdots & g_n \\ 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ 0 & & \ddots & 0 \\ \vdots & & & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix}$$

is the coefficient of $\hat{\varsigma}_1(k)$.

Stability Analysis

In this section, the boundedness of the system is shown where functions are approximated by using CNNs. The aim of neural network saturation compensator (NNSC) is to design proper control law so that the control signal $u(k)$ in equation (18) and weight update law (29) provide the guaranteed tracking performance of overall closed loop system in the presence of saturation nonlinearity. In equation (28) $\varepsilon(k)$ is assumed to be zero.

Theorem 1: Given the system in (27) and Assumption 1 & 2, choose tracking control law (12), the saturation compensator (18) & (20), the estimated NN weights be provided by NN tuning algorithm

$$\hat{w}(k+1) = \left[d(k) + \frac{2\alpha(K-H)r(k)P\hat{\varsigma}_1(k)}{d(k)} + \sqrt{\alpha}r(k) \right] \tag{26}$$

where $d(k)$ is defined as,

$$d(k) = \hat{w}(k) + 2\alpha(K-H)r(k)\varphi(x(k)) \tag{27}$$

By properly selecting the control gains and the design parameters, the tracking error and the NN weights are UUB.

Proof:

Choose the Lyapunov function candidate as

$$J = r^T(k)r(k) + \frac{1}{\alpha}tr(\tilde{w}^T(k)S^{-1}\tilde{w}(k)) \tag{28}$$

where, S^{-1} is a positive definite matrix and α is a constant.
The first difference is given by

$$\Delta J = r^T(k+1)r(k+1) - r^T(k)r(k) + \frac{1}{\alpha} tr(\tilde{w}^T(k+1)\tilde{w}(k+1) - \tilde{w}^T(k)S^{-1}\tilde{w}(k)) \tag{29}$$

Substituting (27) in (32),

$$\begin{aligned} \Delta J \leq & |\hat{w}(k).\varphi(x(k))|^2 + (K-H)^2|r(k)|^2 + P^2|\hat{\varsigma}_1(k)|^2 \\ & - 2\hat{w}^T\varphi(x(k))(K-H)r(k) + 2\hat{w}^T\varphi(x(k))P\hat{\varsigma}_1(k) \\ & - 2(K-H)r(k)P\hat{\varsigma}_1(k) - r^T(k)r(k) + \frac{1}{\alpha}(\|\hat{w}(k+1)\|_F^2 - \|\hat{w}(k)\|_F^2) \end{aligned} \tag{30}$$

Now collecting terms together, adding & subtracting some terms for completing the square yields

$$\begin{aligned} \Delta J \leq & \|\hat{w}(k+1)\|_F^2 + \left[\left\{d(k) + \frac{2\alpha(K-H)r(k)P\hat{\varsigma}_1(k)}{d(k)}\right\} + \sqrt{\alpha}r(k)\right]^2 \\ & - 2\left[d(k) + \frac{2\alpha(K-H)r(k)P\hat{\varsigma}_1(k)}{d(k)}\right]\sqrt{\alpha}r(k) - \alpha[\hat{w}(k)\varphi(x(k)) \\ & + \{(K-H)r(k) + P\hat{\varsigma}_1(k)\}]^2 - [2\alpha(K-H)r(k)\varphi(x(k))]^2 \\ & - 2\alpha\hat{w}^T(k)\varphi^T(x(k))P\hat{\varsigma}_1(k) \end{aligned} \tag{31}$$

with the tuning rule (29), ΔJ is guaranteed to remain negative as long as

$$r(k) > Z(k) \tag{32}$$

and

$$m(k) > 0 \tag{33}$$

where

$$Z(k) = -\frac{\alpha(k)d^T(k)}{2\alpha(K-H)P\hat{\varsigma}_1(k)} \tag{34}$$

$$m(k) = 2\alpha\hat{w}^T(k)\varphi^T(x(k))P\hat{\varsigma}_1(k) \tag{35}$$

Then the tracking error $r(k)$ and the error in weight estimates, $\hat{w}(k)$ are UUB, with the bounds specifically given by in (35)-(38) for the case of theorem 1. This demonstrates that the tracking error $r(k)$ is bounded for all and it remains to show that the weight estimates, $k \geq 0$, are bounded.

Simulation Results

Let us consider a discrete-time nonlinear system

$$
\begin{aligned}
x_1(k+1) &= f_1(x_1(k)) + 0.3 * x_2(k) \\
x_2(k+1) &= f_2(x_2(k)) + \Gamma(k) \\
f_1(x_1(k)) &= 1.4\frac{x_1^2(k)}{1+x_1^2(k)} \\
f_2(x_2(k)) &= \frac{x_1(k)}{1+x_1^2(k)+x_2^2(k)}
\end{aligned}
\tag{36}
$$

where $x_i(k)$ are the state-variables, system output and input respectively.

Here, $x_i(k) = [x_1(k), x_2(k) \dots x_i(k)]^T \in R^i,\ i = 1, \dots, n,\ y(k) \in R$ and $\Gamma(k) \in R,\ \Gamma(k) = A * \tanh(u(k))$, the $u(k)$ is the unconstrained output of backstepping controller.

In this simulation $A = 5$, $\alpha = 0.005$, $K = [.2\,0; 0\,.2]$, $H = [.3\,0; 0\,0]$, $P = [0\,1; 0\,0]$. Initial conditions are $[0\ 0]$ and desired trajectory is given by

$$
x_{1d}(k) = 50 * \left[\sin\left(\frac{k\pi}{20}\right) + \sin\left(\frac{k\pi}{10}\right)\right] \tag{37}
$$

The control input signal $\Gamma(k)$ with the saturation compensator is depicted in Fig.4, the tracking error is shown in Fig.5 and the tracking between the system output & desired output is as shown in Fig.6. From the simulation results, it is clear that the proposed scheme can effectively compensate the saturation nonlinearity in nonlinear systems. Note also that after some initial time that is required for NN to learn the unknown saturation nonlinearity, the NN saturation compensator tracks the desired signal closely and effectively prevents the control signal from reaching saturation limits.

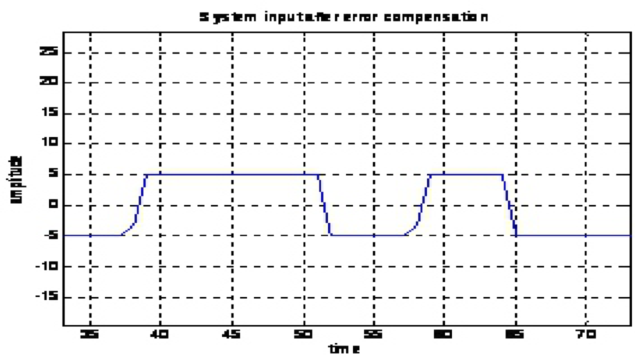

Figure 4. Control signal with Saturation Compensator

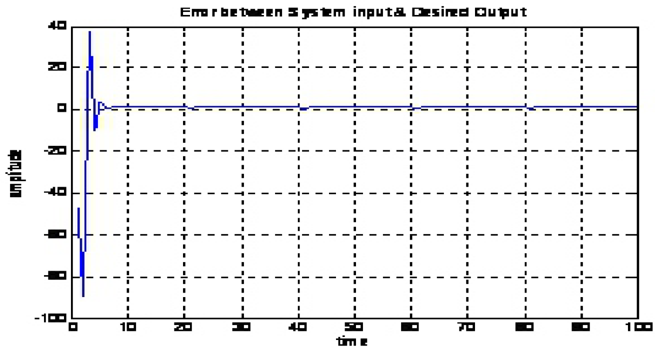

Figure 5. Error between System output & desired response.

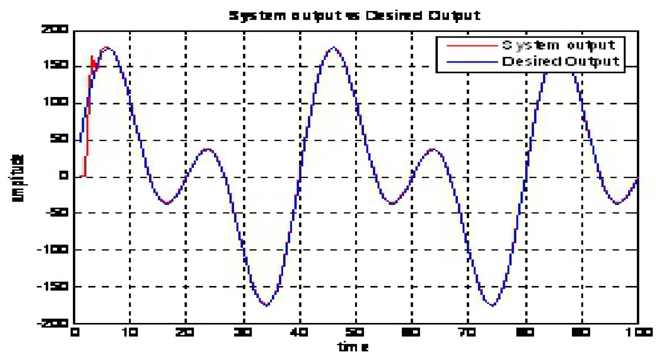

Figure 6. Tracking between System output & Desired response

Conclusion

In this paper CNN saturation compensator is designed for discrete-time nonlinear system in the presence of saturation nonlinearity. CNN is used to approximate the unknown nonlinear functions and the function. The designed intelligent controller based on backstepping technique does not require a saturation model to be known. The neural network controller learns the saturation nonlinearity and updates the weights to prevent the control signal from being saturated. New weight update laws are proposed with the help of Lyapunov theory. The proposed backstepping control guarantees the stability of closed loop system.

References

[1] F. Wu, Z. Lin, Q. Zheng, "Robust Output feedback stabilization of linear system with actuator saturation," IEEE Transactions on Automatic Control, vol. 2, pp. 122–128, 2007.

[2] T. Hu, A. R. Teel, L. Zaccarian, "Stability and performance for saturated systems via quadratic and non quadratic Lyapunov functions," IEEE Transactions on Automatic Control, vol. 51, pp. 1770–1785, 2006.

[3] J. M. Gomes da Silva Jr., S. Tarbouriech, "Local stabilization of discrete-time linear systems with saturating controls: an LMI-based approach," IEEE Transactions on Automatic Control, vol. 46, pp. 119–125, January 2001.

[4] J. M. Gomes da Silva Jr., S. Tarbouriech, "Anti-windup design with guaranteed regions of stability for discrete time linear systems," System and Control Letters, vol. 55, pp. 184–192, 2006.

[5] Wei Lin, "Input Saturation and Global Stabilization of Nonlinear Systems via State & Output Feedback," IEEE Transactions on Automatic Control, vol. 40, pp. 776–782, April 1995.

[6] Federico Morabito, Andrew R. Teel, and Luca Zaccarian, "Nonlinear Antiwindup Applied to Euler-Lagrange Systems," IEEE Transactions on Robotics and Automation, vol. 20, pp. 526–537, June 2004.

[7] Wenzhi Gao and Rastko R. Selmic, "Neural Network Control of a Class of Nonlinear Systems With Actuator Saturation," IEEE Transactions on Neural Networks, vol. 17, No. 1, pp. 147–156, January 2006.

[8] Ajay Kukarni, S. Purwar, "Wavelet based adaptive backstepping controller for a class of nonregular systems with input constraints," Expert systems with applications, vol. 36, No.3, pp. 6686–6696, 2009.

[9] Weisheng Chen, "Adaptive NN control for discrete-time pure-feedback systems with unknown control direction under amplitude and rate actuator constraints," Science Direct, ISA Transactions, vol. 48, pp. 304–311, 2009.

[10] Pingan He and S. Jagannathan, "Reinforcement Learning Neural-Network-Based Controller for Nonlinear Discrete-Time Systems with Input Constraints," IEEE Transactions on Systems, Man, and Cybernetics, vol. 37, No. 2, pp. 425–436, April 2007.

[11] Shuzhi Sam Ge, Chenguang Yang, Tong Heng Lee, "Adaptive robust control of a class of nonlinear strict-feedback discrete-time systems with unknown control directions," System & control letters, pp. 888–895, 2008.

[12] S. S. Ge, G. Y. Li, T. H. Lee, "Adaptive NN control for a class of strict-feedback discrete-time nonlinear systems," Automatica, vol. 39, pp. 807–819, 2003.

[13] Ge S S, Yang C, Lee T H, "Adaptive predictive control using neural network for a class of pure-feedback systems in discrete-time," IEEE Transactions on Neural Networks, vol. 19, No. 9, pp. 1599–613, 2008.

[14] Cheng-Chi Wang, Neng-Sheng Pai, Her-Terng Yau, "Chaos control in AFM system using sliding mode control by backstepping design," Communications in Nonlinear Science and Numerical Simulation, vol. 15, pp. 741–751, March 2010.

[15] Namatame and N. Ueda, "Pattern classification with Chebyshev Neural Network," Int. J. Neural Network, vol. 3, pp. 23–31, 1992.

[16] T. T. Lee, & J. T. Jeng, "The Chebyshev polynomial based unified model neural networks for functions approximations," IEEE Transactions on. Systems, Man & Cybernetics, Part B, vol. 28, No. 6, pp. 925–935, 1998.

[17] S. Purwar, I. N. Kar, A. N. Jha, "Adaptive output feedback tracking control of robot manipulators using position measurements only," Expert systems with applications, vol. 34, pp. 2789–2798, 2008.

[18] Chiman Kwan and F. L. Lewis, "Robust Backstepping Control of Nonlinear Systems Using Neural Networks," IEEE Transactions on Systems, Man, and Cybernetics, vol. 30, No. 6, pp. 753–766, November 2000.