Abstract – A compensation scheme is presented to compensate the effect of dead-zone nonlinearity in a class of uncertain discrete-time nonlinear systems. Chebyshev Neural Network (CNN) is utilized to compensate the dead-zone nonlinearity and the unknown nonlinear functions are also approximated. The control design is attained by introducing dead-zone nonlinearity and using it in the controller design with backstepping technique. Rigorous proofs of closed loop stability for the dead zone compensator are provided which yield tuning algorithm for the weights of both the Neural Networks (NNs). The technique provides a general procedure to guarantee Uniform Ultimate Boundedness (UUB) for all signals in closed loop system.

Index Terms—Backstepping controller, Chebyshev Neural Network (CNN), dead-zone nonlinearity, Lyapunov stability.

I. INTRODUCTION

The dead zone nonlinearity affects the system performance and it is quite common in actuators in real time applications. Being non-smooth and time variant, the dead zone parameters create a challenging problem in control systems. Dead-zone commonly affects all practical systems, such as mechanical, electrical, electronics etc. The study of adaptive control for systems related to dead-zone actuators was initiated in [1], [2] & [3]. The actual controller output can be obtained with the help of fictitious controller [4]-[5] by a recursive backstepping process. Adaptive backstepping design was proposed for feedback linearizable systems with over parameterization in [6] which was eliminated by introducing tuning functions [7].

In motion control systems, dead zone compensator was designed using adaptive fuzzy approach by F.L.Lewis et al [8]. Salim Ibrir et al [9] developed new adaptive compensation algorithm to reduce the effect of dead-zone without dead-zone inverse. The proposed adaptive scheme required only bounds of dead-zone slopes and treated time varying input as a system uncertainty.

T. P. Zhang et al [10] presented an algorithm to avoid explosion of complexity from the traditional backstepping by using Dynamic Surface Control (DSC). Here, an adaptive dynamic surface control of nonlinear system with unknown dead-zone in pure feedback form was developed. Zhonghua Wang et al [11] proposed neural adaptive control for a class of nonlinear systems with unknown dead-zone. They developed a dead-zone compensator and used it in backstepping design by making use of neural networks.

D. A. Racker et al [12] developed an indirect adaptive output feedback control result which applies to discrete-time systems containing a dead-zone nonlinearity and linear system dynamics. An adaptive control approach has been proposed in [13], [14] to address discrete time dead-zone compensation. Through fuzzy logic controller, J.Campos et al [15] proposed a dead zone compensator inverse approach to control the nonlinear discrete-time systems. Direct adaptive regulation of discrete-time nonlinear systems with arbitrary nonlinearities by backstepping was proposed by R. Ordonez [16]. A neuro control of nonlinear discrete-time systems with dead zone and input constraints was given by He. PINGAn et al [17]. In his approach the NN controller consists of two NNs: the first one for compensating the unknown dead-zones; and the second one for compensating the uncertain nonlinear system dynamics. He and Jagannathan [18] proposed an adaptive NN controller using reinforcement learning for a class of discrete-time nonlinear systems. Guo Wencheng et al [19] proposed a concept of time varying dead-zone for a class of discrete-time nonlinear systems by using direct adaptive fuzzy approach. Xin Zhang et al [20] recently presented the work on nonsymmetric dead-zone input for nonlinear systems. Neural network based reinforcement learning controller was developed by two NNs critic & action NNs.

This paper proposes a compensator design to compensate the dead zone nonlinearity in a class of uncertain discrete-time nonlinear systems with backstepping control. In this approach unknown nonlinear functions and dead-zone nonlinearity are approximated with the help of CNN. New weight update law is derived to improve the adaptiveness and convergence property of the system. Uniform Ultimate Boundedness (UUB) is guaranteed, in this scheme, for all signals in closed loop system.

The organization of this paper is as follows: CNN structure and its properties are given in Section-II. The controller design of proposed scheme is presented in Section-III. Section-IV presents the stability analysis and simulation results are presented in Section-V. The conclusions of the paper are given in Section VI.

II. 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 CNN architecture, 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 [21-23]. The Chebyshev polynomials can be obtained by a recursive formula

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

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

Dead-zone characteristic $DZ(k)$ is characterized as:

$$DZ(k) = \begin{cases} m_r (u(k) - b_r) & \text{if } u(k) \geq b_r \\ 0 & \text{if } b_l < u(k) < b_r \\ m_l (u(k) - b_l) & \text{if } u(k) \leq b_l \end{cases} \quad (4)$$

where $u(k)$ is the input and $DZ(k)$ is the output of dead-zone. The parameters $m_r > 0$, $m_l > 0$, $b_r > 0$, and $b_l < 0$ are depicted in Fig.2.

**B. System Dynamics:**
Consider the following discrete-time single-input single-output (SISO) system in strict-feedback form [11]:

$$\xi_i(k + 1) = f_i(\xi_i(k)) + g_i(\xi_i(k))\xi_{i+1}(k) \quad i = 1, 2, 3, \ldots, n - 1$$

$$\xi_n(k + 1) = f_n(\xi_n(k)) + g_n(\xi_n(k))DZ(k)$$

$$y(k) = \xi_1(k) \quad (5)$$

where $\xi_i(k) = [\xi_1(k), \xi_2(k), \ldots, \xi_i(k)]^T \in \mathbb{R}^i$, $i = 1, 2, \ldots, n$, are the state variables, $DZ(k) \in \mathbb{R}$ is the system input and $y(k) \in \mathbb{R}$ is the system output. The smooth function $g_i(\xi_i(k))$ is to be known and $f_i(\xi_i(k))$ is unknown. We have assumed $g_i(\xi_i(k))$ and $g_n(\xi_n(k))$ to be positive. The objective is to design control $u(k)$ to make the system output $y(k)$ to follow a known and bounded trajectory $\xi_{id}(k)$ in the presence of dead-zone nonlinearity $DZ(k)$. Here $u(k)$ is the unconstrained output of backstepping controller after compensating the dead-zone nonlinearity.

**C. Backstepping Controller Design:**
Following the design procedure of backstepping [5], one can define the following error variables $\alpha_i(k)$, where $i = 1, 2, \ldots, n$.

$$\alpha_1(k) = \xi_1(k) - \xi_{1d}(k)$$

$$\alpha_1(k + 1) = \xi_1(k + 1) - \xi_{1d}(k + 1)$$

$$\alpha_2(k) = \xi_2(k) - \xi_{2d}(k)$$

$$\alpha_2(k + 1) = \xi_2(k + 1) - \xi_{2d}(k + 1)$$

$$\alpha_n(k) = \xi_n(k) - \xi_{nd}(k)$$

$$\alpha_n(k + 1) = \xi_n(k + 1) - \xi_{nd}(k + 1) \quad (6)$$

The CNN functional expansion of the input increases the dimension of input pattern. Thus, with the help of CNN, it is very simple to approximate a complex nonlinear system.

**III. CONTROLLER DESIGN**

**A. Dead-zone Nonlinearity**

$$DZ(k)$$

$$\begin{array}{c}
\xi_1 \\
\xi_2 \\
\mathcal{F} \\
\mathcal{E} \\
\mathcal{\Sigma} \\
\hat{f}(\xi)
\end{array}$$

**Fig.1 Chebyshev Neural Network**

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

$$\hat{f}(\xi) = \sigma^T \Psi \quad (2)$$

where $\sigma$ are the estimated weights of neural network. Based on CNN approximation properties [21], [22] & [23], there exist ideal weights $\sigma$, such that the function $f(\xi)$ to be approximated, can be represented as

$$f(\xi) = \sigma^T \Psi + \varepsilon \quad (3)$$

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

The CNN functional expansion of the input increases the dimension of input pattern. Thus, with the help of CNN, it is very simple to approximate a complex nonlinear system.
The biggest problem that is encountered in discrete—time domain in the designing of a backstepping controller is the causality contradiction. Design an ideal fictitious controller by substituting the values of (5) in (6).

\[ \xi_{2d}(k) = 1/g_1(\xi_1(k))[-\hat{f}_1(\xi_1(k)) + \xi_{1d}(k + 1) - k_1a_1(k)] \]

to stabilize the equation corresponding to \( i = 1 \) in (5). On the same lines another fictitious controller can be constructed as,

\[ \xi_{3d}(k) = 1/g_2(\xi_2(k))[-\hat{f}_2(\xi_2(k)) + \xi_{2d}(k + 1) - k_2a_2(k)] \]

and that, for \( n^{th} \) order system

\[ \xi_{(n-1)d}(k) = 1/g_{n-2}(\xi_{n-2}(k))[-\hat{f}_{n-2}(\xi_{n-2}(k)) + \xi_{(n-2)d}(k + 1) - k_{n-2}a_{n-2}(k)] \] (7)

to stabilize the equation corresponding to \( i = n - 1 \) in (5). But in (7) \( \xi_{(n-2)d}(k + 1) \) is a fictitious control for future. This shows that the fictitious controller \( \xi_{(n-1)d}(k) \) is infeasible in practice. If this process is continued to obtain final controller \( DZ(k) \), it is not feasible in practice because it contains more future information. This problem is avoided by approximating the future values using CNN.

Substituting (5) and \( \xi_{id}(k + 1) \) from (7) in (6), the error variables are obtained as follows,

\[ \alpha_1(k + 1) = \hat{f}_1(k) + \epsilon_1(k) + g_1(\xi_2(k))\xi_1(k) - k_1a_1(k) \]

\[ \alpha_n(k + 1) = \hat{f}_n(k) + \epsilon_n(k) + g_n(\xi_n(k))\beta_1(k) - k_n\alpha_n(k) \] (8)

where functional estimation errors are

\[ \hat{f}_1(k) = f_1(k) - \hat{f}_1(k) \]
\[ \hat{f}_2(k) = f_2(k) - \hat{f}_2(k) \]

\[ \hat{f}_n(k) = f_n(k) - \hat{f}_n(k) \] (9)

and \( \beta_1(k) \) is defined in (18).

Since dead-zone nonlinearity is being used in the controller design with backstepping technique, the output of backstepping has the effect of dead-zone nonlinearity in (5). The control signal \( DZ(k) \) is zero within the range of dead-zone bounds as shown in (4) and, within this range, the control signal is not fully implemented by the device, denoted by \( \beta(k) \).

\[ \beta(k) = u(k) - DZ(k) \] (10)

and \( DZ(k) = u(k) + \beta(k) \)

Let \( u(k) + \beta(k) = u'(k) \) (11)

Similar to (7), constrained output of controller can be obtained,

\[ u'(k) = 1/g_n(\xi_n(k))[-\hat{f}_n(\xi_n(k)) + \xi_{nd}(k + 1) - k_n\alpha_n(k)] \] (12)

The dead-zone inverse is used to compensate the effect of dead-zone nonlinearity in the nonlinear discrete-time system in (5). The actuator dead-zone described in (10), which is not implemented by the device, is unknown. In this paper unknown nonlinearity \( \beta(k) \) is approximated by using Chebyshev neural network (CNN).

\[ \beta(k) = v^T\Psi(\bar{x}) \] (13)

where \( v \) is the weight and \( \bar{x} = [\xi_{1d} \ldots]^T \) are the inputs of neural network compensator. The parameter \( s \) is tracking error vector as defined in (20). The approximation of unknown nonlinearity \( \beta(k) \) is given by

\[ \hat{\beta}(k) = \theta^T\Psi(\bar{x}) \] (14)

Here \( \theta \) is obtained as per tuning rule in (23). The unconstrained output of the controller is obtained by subtracting the approximated unknown nonlinearity using CNN in (14) from constrained output of the controller in (12).

\[ u(k) = u'(k) - \hat{\beta}(k) \] (15)

**IV. STABILITY ANALYSIS**

The boundedness of the system is proved where functions are approximated by using Chebyshev neural networks (CNNs). To make controller more adaptive, effective weight tuning law is required which is derived with the help of Lyapunov theory.

The functional estimation error in (9) can be represented in terms of neural network for the value of ideal weight \( \sigma_i(k) \), where \( i = 1,2, \ldots, n \) [4].

\[ \hat{f}_i(\xi_i(k)) = \sigma_i(k)\Psi_i(\xi_i(k)) + \epsilon_i(k) \] (16)

where \( \epsilon_i(k) \) is the reconstrcutonal error as defined in (20) & \( \sigma_i \) is the weight of CNN.

Substituting (16) in (8), error variables are obtained as follows:

\[ \alpha_1(k + 1) = \hat{\sigma}_1(k)\Psi_1(\xi_1(k)) + \epsilon_1(k) + g_1(\xi_2(k))\xi_1(k) - k_1a_1(k) \]

\[ \alpha_n(k + 1) = \hat{\sigma}_n(k)\Psi_n(\xi_n(k)) + \epsilon_n(k) + g_n(\xi_n(k))\beta_1(k) - k_n\alpha_n(k) \] (17)

where $\hat{\beta}_1(k) = \beta(k) - \hat{\beta}(k)$ \hspace{1cm} (18)

The equation (17) can be re-written in as,

\[
s(k + 1) = \hat{\sigma}(k)\Psi(\xi(k)) + \varepsilon(k) + Ns(k) + Q\hat{\beta}(k) - Ks(k) + \mu(k)
\]

where \[s(k) = [a_1^T(k) \ a_2^T(k) \ \ldots \ a_{n+1}^T(k)]^T\]

\[
s(k + 1) = [a_1^T(k + 1) \ a_2^T(k + 1) \ \ldots \ a_{n+1}^T(k + 1)]^T
\]

\[
\hat{\sigma}(k) = diag(\hat{\sigma}_1(k) \ \hat{\sigma}_2(k) \ \ldots \ \hat{\sigma}_n(k))
\]

\[
\varepsilon(k) = [\varepsilon_1(k) \ \varepsilon_2(k) \ \ldots \ \varepsilon_n(k)]^T
\]

\[
\Psi(\xi(k)) = [\Psi_1^T(\xi(k)) \ \Psi_2^T(\xi(k)) \ \ldots \ \Psi_n^T(\xi(k))]^T
\]

\[K = diag(k_1 \ k_2 \ \ldots \ k_n) \ \& \ \mu \text{ is the robustifying term defined in (22).}\]

\[
N = \begin{bmatrix}
0 & g_1 & 0 & \ldots & 0 & 0 \\
0 & 0 & g_2 & \ldots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & g_i & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & 0 & g_n \\
\end{bmatrix}
\]

\[
Q = \begin{bmatrix}
0 & \ldots & \ldots & g_n \\
0 & \ldots & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & \ldots & \ldots & 0 \\
0 & \ldots & \ldots & 0 \\
\end{bmatrix}
\]

The value of $\varepsilon(k)$ in (20) is assumed to be zero, and $K - N = K_m$, equ. (19) reduces to

\[
s(k + 1) = \hat{\sigma}(k)\Psi(\xi(k)) - K_m s(k) + Q\hat{\beta}(k) + \mu(k)
\]

Theorem 1: Given the system in (21), choose tracking control law in (12), the dead-zone compensator in (14) & (15) and the robustifying term [24] as

\[
\mu(k) = -f_m(\xi(k))Sgn(s(k))
\]

where $f_m(\xi(k))$ is the bound on the functional estimation error and $Sgn(s(k))$ is a standard sign function. Let the estimated NN weights be provided by the NN tuning algorithm

\[
\hat{\sigma}(k + 1) = \frac{1}{\sqrt{\eta}}[\hat{\sigma}(k)\eta - \Omega|\Psi(\xi(k))|\left(\hat{\beta}(k) + \mu(k) - K_m s(k)\right) + \sqrt{\Omega}s(k)]
\]

where $\eta = 1 - \Omega|\Psi(\xi(k))|^2$ and $\Omega$ is a constant parameter. By properly selecting the control gains & the design parameters, the filtered error $s(k)$ and NN weights $\hat{\sigma}(k)$ are UUB.

Proof: Consider the Lyapunov function

\[
L = s^T(k) s(k) + \frac{1}{n} tr(\hat{\sigma}^T(k)\hat{\sigma}(k))
\]

The difference equation.

\[
\Delta L = L(k + 1) - L(k)
\]

\[
\Delta L = s^T(k + 1) s(k + 1) - s^T(k) s(k) + \frac{1}{n} tr(\hat{\sigma}^T(k + 1) \hat{\sigma}(k + 1) - \hat{\sigma}^T(k) \hat{\sigma}(k))
\]

Substituting (21) in (28),

\[
\Delta L = [\hat{\sigma}(k)\Psi(\xi(k)) - K_m s(k) + Q\hat{\beta}(k) + \mu(k)]\left[\hat{\sigma}(k)\Psi(\xi(k)) - K_m s(k) + Q\hat{\beta}(k) + \mu(k)\right]^T
\]

\[
s^T(k) s(k) + \frac{1}{n} tr(\hat{\sigma}^T(k + 1) \hat{\sigma}(k + 1) - \hat{\sigma}^T(k) \hat{\sigma}(k))
\]

(27)

Assumption 1: (Bounded Ideal NN Weights): The ideal NN weights $\sigma$ are bounded so that $||\sigma||_F \leq \sigma_M$, with $\sigma_M$ a known bound. The symbol $||A||_F$ denotes the Frobenius norm, i.e. given a matrix $A$, the Frobenius norm is given by [10],

\[
||A||_F^2 = tr(A^T A)
\]

with the Assumption 1 in (28), (27) can be write

\[
\Delta L = \{\hat{\sigma}(k)\Psi(\xi(k)) - K_m s(k) + Q\hat{\beta}(k) + \mu(k)\}^T \{\hat{\sigma}(k)\Psi(\xi(k)) - K_m s(k) + Q\hat{\beta}(k) + \mu(k)\} - \frac{1}{n} tr(\hat{\sigma}(k + 1)\hat{\sigma}(k + 1) - \hat{\sigma}(k)\hat{\sigma}(k))
\]

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

\[
\Delta L \leq -||\hat{\sigma}(k + 1)||^2 + \frac{1}{\sqrt{n}}[\hat{\sigma}(k)\eta - \Omega|\Psi(\xi(k))|\left(\hat{\beta}(k) + \mu(k) - K_m s(k)\right) + \sqrt{\Omega}s(k)]^2
\]

\[
- \frac{2\sqrt{\Omega}s(k)}{\sqrt{n}} [\hat{\sigma}(k)\eta - \Omega|\Psi(\xi(k))|\left(\hat{\beta}(k) + \mu(k) - K_m s(k)\right)] - \frac{\eta}{2} [\hat{\sigma}(k)\eta - \Omega|\Psi(\xi(k))|\left(\hat{\beta}(k) + \mu(k) - K_m s(k)\right)]^2
\]

\[
- \Omega(K_m s(k) - Q\beta(k) + \mu(k))^2
\]

(30)

with tuning algorithm (23), $\Delta L$ is guaranteed to remain negative as long as

\[
s(k) > \frac{1}{K_m} [Q\hat{\beta}(k) + \mu(k) - \{\hat{\sigma}(k)\eta\}^2]\]

\[
\frac{\eta}{2} [\hat{\sigma}(k)\eta - \Omega|\Psi(\xi(k))|\left(\hat{\beta}(k) + \mu(k) - K_m s(k)\right)]^2
\]

(32)

\[
\Omega(K_m s(k) - Q\beta(k) + \mu(k))^2 > 0
\]

(33)

Then the tracking error $s(k)$ and the weight estimates $\hat{\sigma}(k)$, are UUB, with the bounds specifically given in (31)-(33) for the case of Theorem1.
V. SIMULATION RESULTS

Let us consider a discrete-time nonlinear system

\[ x_1(k+1) = f_1(x_1(k)) + 0.3 \cdot x_2(k) \]
\[ x_2(k+1) = f_2(x_2(k)) + DZ(k) + d(k) \]

\[
\begin{align*}
  f_1(x_1(k)) &= 14 \frac{x_1^2(k)}{1 + x_1^2(k)} \\
  f_2(x_2(k)) &= \frac{x_2}{1 + x_1^2(k) + x_2^2(k)}
\end{align*}
\]
\[ d(k) = 0.1 \cos(0.05k) \cos(x_1(k)) \]  

where \( x_1(k) = [x_1(k), x_2(k), ..., x_n(k)]^T \in \mathbb{R}^n \), \( i = 1, ..., n \), are the state-variables, \( y(k) \in \mathbb{R} \) is the system output and \( u(k) \in \mathbb{R} \) is system input, \( d(k) \) denotes the external disturbance.

\( DZ(k) \) is already defined in (4). In the simulation, positive and negative slopes \( m_r = 1 \) & \( m_l = 1 \), \( b_r = 100 \) & \( b_l = -100 \), \( \Omega = 0.5 \), \( K = [0.01 \ 0 \ 0; 0.01 \ 0; 0 \ 0 \ 1.2] \), \( N = [0 \ 0.3 \ 0; -0.3 \ 0 \ 0] \), \( Q = [0 \ 0] \). Initial conditions are \([0 \ 0]\) and desired trajectory is given by

\[ \xi_{1d}(k) = 50 \cdot [\sin(\frac{k\pi}{20}) + \sin(\frac{k\pi}{10})] \]  

The control input signal \( DZ(k) \) with the dead-zone compensator is depicted in Fig.3. The tracking error is shown in Fig.4 and tracking between the system output and desired response is as shown in Fig.5. From the simulation results, it is clear that the proposed scheme can effectively compensate the dead-zone nonlinearity by CNN compensator in nonlinear systems. After some initial time, CNN learns the unknown dead-zone nonlinearity and the compensator closely tracks the desired signal preventing the control signal from dead-zone limits.

VI. CONCLUSIONS

A Chebyshev Neural Network (CNN) pre-compensator has been proposed for dead-zone compensation in uncertain discrete-time nonlinear system with backstepping. In this paper, new weight update law is derived with the help of Lyapunov theory. The proposed backstepping technique is robust and guarantees the stability of closed loop system. The tracking error is reduced as much as possible by updating the design parameters. The simulation results are presented to show the effectiveness of the proposed scheme which reveals that the tracking error of the system is minimized as compared to the previous results.

REFERENCES


