BACKLASH COMPENSATION IN NONLINEAR SYSTEMS USING DYNAMIC INVERSION BY NEURAL NETWORKS

Rastko R. Šelmic and Frank L. Lewis

Automation and Robotics Research Institute
The University of Texas at Arlington
7300 Jack Newell Blvd. South
Fort Worth, Texas 76118-7115
tel. 817-272-5957, fax. 817-272-5989
Email: rselmic@arrirs04.uta.edu
flewis@controls.uta.edu

Submitted to:

Asian Journal of Control

October 1, 1999

Keywords: neurocontrol, neural networks, backlash compensation, actuator nonlinearity control, dynamic inversion.

Research supported by NSF Grant ECS-9521673, and Texas ATP Grant 003656-027, and ARO Grant 39657-MA.

ABSTRACT

A dynamic inversion compensation scheme is presented for backlash. The compensator uses the backstepping technique with neural networks (NN) for inverting the backlash nonlinearity in the feedforward path. The technique provides a general procedure for using NN to determine the dynamic preinverse of an invertible dynamical system. A tuning algorithm is presented for the NN backlash compensator which yields a stable closed-loop system.

1 INTRODUCTION

A general class of industrial motion control systems has the structure of a nonlinear dynamical system preceded by some *nonlinearities* in the actuator, either deadzone, backlash, saturation, etc. This includes xy-positioning tables [19], robot manipulators [14], overhead crane mechanisms, and more. The problems are particularly exacerbated when the required accuracy is high, as in micropositioning devices. Due to the nonanalytic nature of the actuator nonlinearities and the fact that their exact nonlinear functions are unknown, such systems present a challenge for the control design engineer. Proportional-derivative (PD) controllers have been observed to result in limit cycles if the actuators have deadzones or backlash. Rigorous results for motion tracking of such systems are notably sparse, though ad hoc techniques relying on simulations for verification of effectiveness are prolific. A neural net scheme for deadzone compensation appears in [13], but no proof of performance is offered. Stability proofs and design of deadzone compensator for industrial positioning systems using a fuzzy logic controller are given in [18].

Recently, in seminal work several rigorously derived adaptive schemes have been given for actuator nonlinearity compensation [36]. Backlash compensation is considered in [35]. Compensation for nonsymmetric deadzones is considered in [34] for linear systems, in [25] for nonlinear systems in Brunovsky form with known nonlinear functions. For the dynamic system in Lagrangian form, deadzone compensation using NN is given in [30],

[31]. It is not required for deadzone to be symmetric, and the function outside the dead-band may not be linear.

Dynamic inversion using NN is presented in [10], [14], [20] where NN is used for cancellation of the inversion error. Modeling inverse dynamics in the feedforward path using recurrent neural networks is shown in [38]. A compensated inverse dynamics approach using adaptive and robust control techniques is presented in [33].

In this paper we assume a general model of backlash which is not required to be symmetric. We use a backstepping approach to derive a compensator which has a neural network in the feedforward loop. This amounts to a dynamic inversion approach. The proposed method can be applied for compensation of a large class of invertible dynamical nonlinearities. We show how to design the backlash compensator, and provide a *rigorous closed-loop system stability proof* that guarantees small tracking error and bounded NN weights. The system here is assumed to be in Brunovsky form. Simulation results show that NN backlash compensator can significantly reduce degrading effect of backlash nonlinearity.

2 BACKGROUND

Let S be a compact simply connected set of $\Re^n$. With map f: $S \rightarrow \Re^m$, define $C(S)$ as the space such that f is continuous. The space of functions whose r-th derivative is continuous is denoted by $C^r(S)$, and the space of smooth functions is $C^\infty(S)$.

By $\|\cdot\|$ is denoted any suitable vector norm. When it is required to be specific we denote the p-norm by $\|\cdot\|_p$. The supremum norm of f(x), over S, is defined as [2]

$$\sup_{x \in S} \|f(x)\|, \quad f: S \rightarrow \Re^m. \tag{2.1}$$

Given A=[a_{ij}], B$\in \Re^{mxn}$ the Frobenius norm is defined by

$$\|A\|_F^2 = tr(A^T A) = \sum_{i,j} a_{ij}^2 \;, \tag{2.2}$$

with tr( ) the trace. The associated inner product is $\langle A, B\rangle_F = tr(A^T B)$. The Frobenius norm is compatible with the 2-norm so that $\|Ax\|_2 \le \|A\|_F \|x\|_2$.

When $x(t) \in \Re^n$ is a function of time we use the standard L_p norms. It is said that x(t) is bounded if its L_∞ norm is bounded. Matrix $A(t) \in \Re^{mxn}$ is bounded if its induced matrix ∞-norm is bounded.

Consider the nonlinear system

$$\dot{x} = g(x, u, t), \quad y = h(x, t) \tag{2.3}$$

with state $x(t) \in \Re^n$. The equilibrium point x_e is said to be uniformly ultimately bounded (UUB) if there exists a compact set $S \subset \Re^n$, so that for all $x_0 \in S$ there exists an ε>0, and a number $T(\varepsilon, x_0)$ such that $\|x(t) - x_e\| \le \varepsilon$ for all $t \ge t_0 + T$. That is, after a transition period T, the state x(t) remains within the ball of radius ε around x_e.

2.1 Background on Neural Networks

NN have been used extensively in feedback control systems. Most applications are ad hoc with no demonstrations of stability. The stability proofs that do exist rely almost invariably on the universal approximation property for NN [3], [17], [26], [27], [28]. NN approximation of the discontinuous functions with application to friction and deadzone compensation is given in [29], [30], [31].

The two-layer NN in Figure 2.1 consists of two layers of tunable weights and has a hidden layer and an output layer. The hidden layer has L neurons, and the output layer has m neurons. The multilayer NN is a nonlinear mapping from input space $\Re^n$ into output space $\Re^m$.

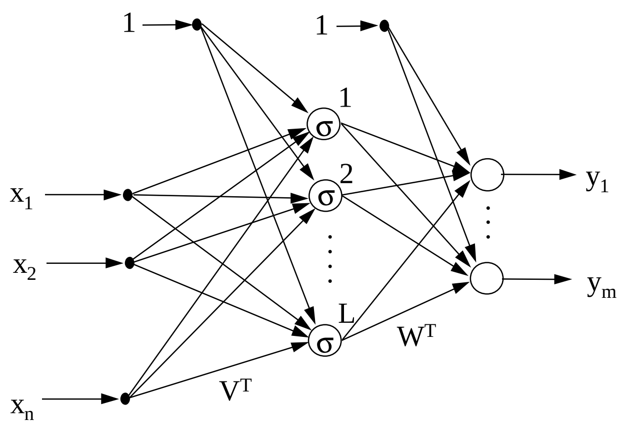

Figure 2.1 Two-layer NN.

The NN output y is a vector with m components that are determined in terms of the n components of the input vector x by the equation

$$y_i = \rho\left(\sum_{\ell=1}^{L} w_{i\ell}\,\sigma\left(\sum_{j=1}^{n} v_{\ell j} x_j + v_{\ell 0}\right) + w_{i0}\right); \quad i=1, 2, ..., m. \tag{2.4}$$

where $\sigma(\cdot)$ are the hidden layer activation functions, $\rho(\cdot)$ are the output layer activation functions and L is the number of hidden-layer neurons. The first-layer interconnection weights are denoted v_{lj} and the second-layer interconnection weights by w_{il}. The threshold offsets are denoted by v_{l0}, w_{i0}.

By collecting all the NN weights v_{lj}, w_{il} into matrices V^T, W^T, the NN equation with linear output activation function $\rho(\cdot)$, may be written in terms of vectors as

$$y = W^T \sigma(V^T x) \ . \tag{2.5}$$

The thresholds are included as the first column of the weight matrices W^T, V^T; to accommodate this, the vector x and $\sigma(\cdot)$ need to be augmented by placing a ‘1’ as their first element (e.g. $x=[1\ x_1\ x_2\ ...\ x_n]^T$). In this equation, to represent (2.4) one has sufficient generality if $\sigma(\cdot)$ is taken as a diagonal function from $\Re^L$ to $\Re^L$, that is $\sigma(z)=\text{diag}\{\sigma(z_l)\}$ for a vector $z=[z_1\ \ z_2\ ...\ z_L]^T \in \Re^L$. For notational convenience we define the matrix of all the weights as

$$Z = \begin{bmatrix} W & \\ & V \end{bmatrix} . \qquad (2.6)$$

There are many different ways to choose the activation functions $\sigma(\cdot)$, including sigmoid, hyperbolic tangent, etc. We will use the sigmoid activation function given by

$$\sigma(x) = \frac{1}{1 + e^{-\beta x}} , \qquad (2.7)$$

Many well-known results say that any sufficiently smooth function can be approximated arbitrarily closely on a compact set using a two-layer NN with appropriate weights. For instance, Cybenko's result [6] for continuous function approximation says that given any function $f \in C(S)$, with S compact subset of $\Re^n$, and any $\varepsilon_N>0$, one has

$$f(x) = W^T \sigma(V^T x) + \varepsilon(x) \;, \qquad (2.8)$$

where the $\varepsilon(x)$ is the NN approximation error, and $\|\varepsilon(x)\| \le \varepsilon_N$ for $x \in S$. Barron has shown [1] that NN can serve as universal approximators for continuous functions with a fundamental lower bound of order $(1/L)^{2/n}$. The approximating weights V and W are ideal target weights, and it is assumed that they are bounded such that $\|V\|_F \le V_M$, $\|W\|_F \le W_M$, or $\|Z\|_F \le Z_M$.

3 BACKLASH NONLINEARITY

Here is assumed a general model of the backlash which is not required to be symmetric. We use a backstepping approach to derive a compensator which has a neural network in the feedforward loop. The proposed method can be applied for compensation of a large class of invertible dynamical nonlinearities.

To focus on backlash compensation, we assume the system is in Brunovsky form. The generality of the method and its applicability to a broad range of nonlinear functions make this approach a useful tool for compensation of backlash, hysteresis, etc. Backlash compensation is done using dynamic inversion, where NN is used for the dynamic inversion compensation [20], [14].

3.1 Backlash Nonlinearity and Backlash Inverse

The backlash nonlinearity is shown in Figure 3.1, and a mathematical model is given by (3.1) [36].

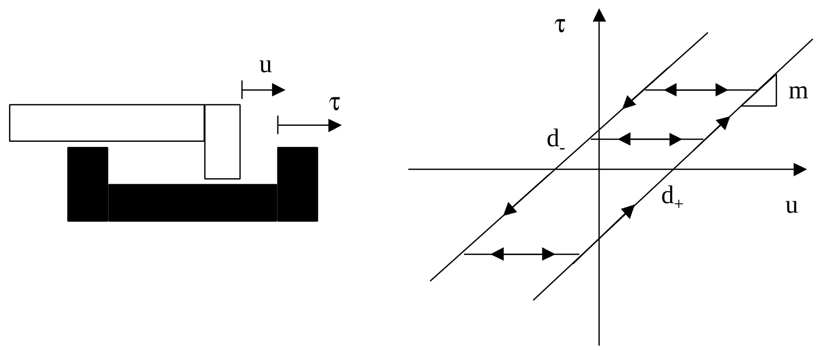

Figure 3.1 Backlash nonlinearity.

$$\dot{\tau} = B(\tau, u, \dot{u}) = \begin{cases} m\dot{u}\,, \text{ if } \dot{u} > 0 \text{ and } \tau = mu - md_+ \\ \quad\;\; \text{if } \dot{u} < 0 \text{ and } \tau = mu - md_- \\ 0\,, \quad \text{otherwise} \end{cases} \tag{3.1}$$

One can see that backlash is a first-order velocity-driven dynamic system, with inputs u and $\dot{u}$, and state τ. It contains its own dynamics, therefore its compensation requires the design of dynamic compensator.

Whenever the motion u(t) changes its direction, the motion $\tau(t)$ is delayed from motion of u(t). The objective of a backlash compensator is to make this delay as small as possible, i.e. to make the $\tau(t)$ to closely follow u(t). In order to cancel the effect of backlash in the system, the backlash precompensator needs to generate the inverse of the backlash nonlinearity. The backlash inverse function is shown in Figure 3.2.

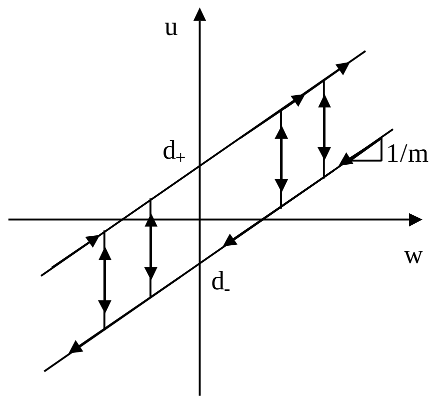

Figure 3.2 Backlash inverse.

The dynamics of the NN backlash compensator is given by

$$\dot{u} = B_{inv}(u, w, \dot{w}) \ . \tag{3.2}$$

The backlash inverse characteristic shown in the Figure 3.2 can be decomposed into two functions: a direct feedforward term plus the additional *modified backlash inverse* term as shown in the Figure 3.3. This decomposition allows design of a compensator that has a better structure when a NN is used in the feedforward path.

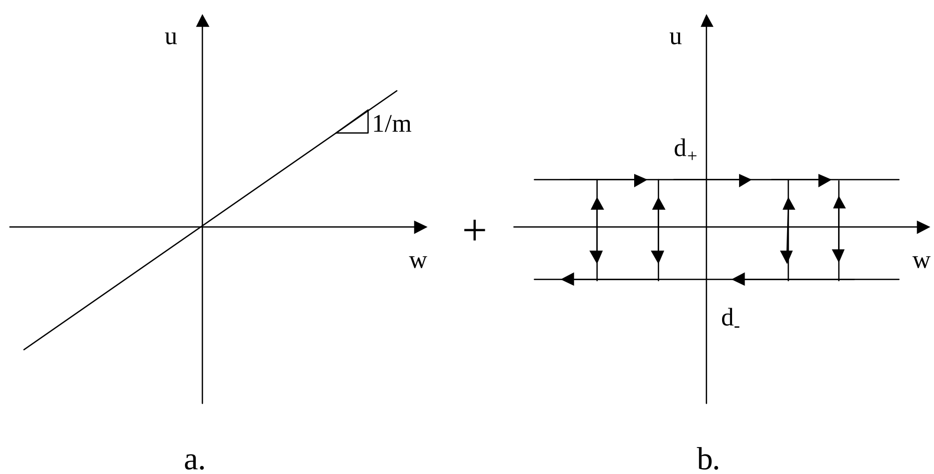

Figure 3.3 Backlash inverse decomposition.

4 NN CONTROLLER WITH BACKLASH COMPENSATION

The NN backlash compensator is designed using the backstepping technique originally developed by Krstic et al [9]. In this section we will show how to tune or learn the weights of the NN on-line so that the tracking error is guaranteed small and all internal states (e.g. the NN weights) are bounded. It is assumed that the actuator output $\tau(t)$ is measurable.

4.1 Dynamics of Nonlinear Motion Systems

The dynamics of a large class of single-input nonlinear systems can be written in the Brunovsky form

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_n &= f(x) + \tau_d + \tau \\ y &= x_1 \end{aligned} \tag{4.1}$$

where the output is y(t), the state is $x \equiv [x_1\ x_2 \dots x_n]^T$, τ_d is a disturbance, $\tau(t)$ is actuator output, and function f(x) represents system nonlinearities like friction, etc. The actuator output $\tau(t)$ is related to the control input u(t) through the backlash nonlinearity (3.1). Therefore overall dynamics of the system consists of (4.1) and backlash dynamics (3.1).

The following assumptions are needed. They are typical assumptions commonly made in the literature and hold for many practical systems.

Assumption 1 (Bounded disturbance). The unknown disturbance satisfies $\|\tau_d\| \le \tau_M$, with $\tau_M(t)$ a known positive constant.

¦

Assumption 2 (Bounded estimation error). The nonlinear function f(x) is assumed to be unknown, but a fixed estimate $\hat{f}(x)$ is assumed known such that the functional estimation error, $\tilde{f}(x) = f(x) - \hat{f}(x)$, satisfies

$$\|\tilde{f}(x)\| \le f_M(x) \ , \tag{4.2}$$

for some known bounding function $f_M(x)$.

¦

This is not unreasonable [4], [14], as in practical systems the bound $f_M(x)$ can be computed knowing the upper bound on payload masses, frictional effects, and so on.

To design a motion controller that causes the system output, y(t), to track a smooth prescribed trajectory, $y_d(t)$, we define the desired state as

$$x_d(t) = \begin{bmatrix} y_d & \dot{y}_d & \dots & y_d^{(n-1)} \end{bmatrix}^T \ , \tag{4.3}$$

with $y_d^{(n-1)}$ the (n-1)-st derivative. We define the *tracking error* by

$$e = x - x_d \ , \tag{4.4}$$

and the *filtered tracking error* by

$$r = \begin{bmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_{n-1} & 1 \end{bmatrix} e \equiv \Lambda^T e \equiv \begin{bmatrix} \Lambda^T & 1 \end{bmatrix} e \ , \tag{4.5}$$

with Λ a gain parameter vector selected so that $e(t) \to 0$ exponentially as $r(t) \to 0$. Then, (4.5) is stable system so that e(t) is bounded as long as controller guaranties that the filtered error r(t) is bounded.

Differentiating (4.5) and invoking (4.1), it can be seen that the dynamics are expressed in terms of filtered error as

$$\dot{r} = f(x) + Y_d + \tau_d + \tau \ , \tag{4.6}$$

where

$$Y_d = -y_d^{(n)} + \begin{bmatrix} 0 & \Lambda^T \end{bmatrix} e \tag{4.7}$$

is known function of the desired trajectory and actual states.

<u>Assumption 3</u> (Bounded desired trajectory). The desired trajectory is bounded so that

$$\|x_d(t)\| \le X_d \ , \tag{4.8}$$

where X_d is a known constant.

4.2 Backstepping Controller Design with NN Backlash Compensation

A robust compensation scheme for unknown terms in f(x) is provided by selecting the tracking controller

$$\tau_{des} = -K_V r - \hat{f}(x) - Y_d + v_1 \quad , \tag{4.9}$$

with $\hat{f}(x)$ an estimate for the nonlinear terms f(x), and $v_1(t)$ a robustifying term to be selected for the disturbance rejection. The feedback gain matrix $K_V>0$ is often selected diagonal. The estimate $\hat{f}(x)$ is fixed in this paper and will not be adapted, as is common in robust control techniques [4], [14]. If f(x) in (4.1) is unknown, it can be estimated using adaptive control techniques [5], [14], or the neural network controller in [17].

The next theorem is the first step in the backstepping design; it shows that the desired control law (4.9) will keep the filtered tracking error small. In Theorem 2 we will later rigorously show how to design the NN controller so the actual control law approaches the desired τ_{des}, still keeping the NN approximation error weights bounded.

Theorem 1 **(Control law for outer tracking loop).**

Given the system (4.1) and Assumptions 1-2, select the tracking control law (4.9). Choose the robustifying signal v_1 as

$$v_1(t) = -(f_M(x) + \tau_M)\frac{r}{\|r\|} \quad . \tag{4.10}$$

Then the filtered tracking error r(t) is UUB and it can be kept as small as desired by increasing the gains K_v.

Proof: Select the Lyapunov function candidate

$$L_1 = \frac{1}{2} r^T r \quad . \tag{4.11}$$

Differentiating L_1 and using (4.6) yields

$$\dot{L}_1 = r^T\left(f(x) + Y_d + \tau_d + \tau_{des}\right) \quad . \tag{4.12}$$

Applying the tracking control law (4.9) one has

$$\dot{L}_1 = r^T\left(f(x) + Y_d + \tau_d - K_V r - \hat{f}(x) - Y_d + v_1\right) \ , \tag{4.13}$$

$$\dot{L}_1 = r^T\left(\tilde{f}(x) + \tau_d - K_V r + v_1\right) \ . \tag{4.14}$$

$$\dot{L}_1 = -r^T K_V r + r^T\left(\tilde{f}(x) + \tau_d - \left(f_M(x) + \tau_M\right)\frac{r}{\|r\|}\right) \ . \tag{4.15}$$

Expression (4.15) can be bounded as

$$\dot{L}_1 \le -K_{V\min}\|r\|^2 - \|r\|(f_M + \tau_M) + \|r\|\left\|\tilde{f} + \tau_d\right\| \ , \tag{4.16}$$

Using Assumptions 1 and 2, one can conclude that $\dot{L}_1$ is guaranteed negative for as long as $\|r\| \neq 0$.

¦

4.3 NN Backlash Compensation using Dynamic Inversion

In the Theorem 1 is given the control law which ensures stability in terms of the filtered tracking error. In the presence of the unknown backlash nonlinearity, the desired and actual value of the control signal τ will be different. Following the idea of dynamic inversion where neural network is used for compensation of the inversion error, originally given by Calise et al. [14], [20], we give a rigorous analysis of the closed-loop system stability.

The actuator output given by (4.9) is the desired, ideal signal. In order to find the complete system error dynamics define the error between the desired and actual actuator outputs as

$$\tilde{\tau} = \tau_{des} - \tau \ . \tag{4.17}$$

Differentiating one has

$$\begin{aligned} \dot{\tilde{\tau}} &= \dot{\tau}_{des} - \dot{\tau} \\ &= \dot{\tau}_{des} - B(\tau, u, \dot{u}) \end{aligned} \tag{4.18}$$

which together with (4.6) and involving (4.9) represent the complete system error dynamics. The goal of the second step in backstepping controller design is to ensure that actual actuator output follows desired actuator output even in the presence of the backlash nonlinearity, thus achieving the backlash compensation.

The dynamics of the backlash nonlinearity can be written as

$$\dot{\tau} = \varphi \tag{4.19}$$

$$\varphi = B(\tau, u, \dot{u}) \tag{4.20}$$

where $\varphi(t)$ is pseudo-control input [14], [20]. In the case of known backlash, the ideal backlash inverse is given by

$$\dot{u} = B^{-1}(u, \tau, \varphi) \ . \tag{4.21}$$

Since the backlash and therefore its inverse are not known, one can only approximate the backlash inverse

$$\dot{\hat{u}} = \hat{B}^{-1}(\hat{u}, \tau, \hat{\varphi}) \ . \tag{4.22}$$

The backlash dynamics can now be written as

$$\begin{aligned} \dot{\tau} &= B(\tau, \hat{u}, \dot{\hat{u}}) \\ &= \hat{B}(\tau, \hat{u}, \dot{\hat{u}}) + \tilde{B}(\tau, \hat{u}, \dot{\hat{u}}) \ , \\ &= \hat{\varphi} + \tilde{B}(\tau, \hat{u}, \dot{\hat{u}}) \end{aligned} \tag{4.23}$$

where $\hat{\varphi} = \hat{B}(\tau, \hat{u}, \dot{\hat{u}})$ and therefore its inverse $\dot{\hat{u}} = \hat{B}^{-1}(\tau, \hat{u}, \hat{\varphi})$. The unknown function $\tilde{B}(\tau, \hat{u}, \dot{\hat{u}})$, which represents the backlash inversion error, will be approximated using neural network.

Based on the NN approximation property, the backlash inversion error can be represented as

$$\tilde{B}(\tau, \hat{u}, \dot{\hat{u}}) = W^T \sigma(V^T x_{nn}) + \varepsilon \ , \tag{4.24}$$

where NN input vector is chosen as $x_{nn} = [1 \ \ r^T \ \ x_d{}^T \ \ \tilde{\tau}^T \ \ \tau]^T$, and ε represents the NN approximation error.

Define $\hat{V}$, $\hat{W}$ as estimates of the ideal NN weights, which are given by the NN tuning algorithms. Define the weight estimation errors as

$$\tilde{V} = V - \hat{V}, \quad \tilde{W} = W - \hat{W}, \quad \tilde{Z} = Z - \hat{Z} \ , \tag{4.25}$$

and the hidden-layer output error for a given x as

$$\tilde{\sigma} = \sigma - \hat{\sigma} \equiv \sigma\left(V^T x_{nn}\right) - \sigma\left(\hat{V}^T x_{nn}\right) . \tag{4.26}$$

In order to design the stable closed-loop system with backlash compensation, one selects a nominal backlash inverse $\dot{\hat{u}} = \hat{\varphi}$ and pseudo-control input $\hat{\varphi}$ as

$$\hat{\varphi} = K_b \tilde{\tau} + \dot{\tau}_{des} - \hat{W}^T \sigma\left(\hat{V}^T x_{nn}\right) + v_2 \ , \tag{4.27}$$

where $v_2(t)$ is a robustifying term detailed later.

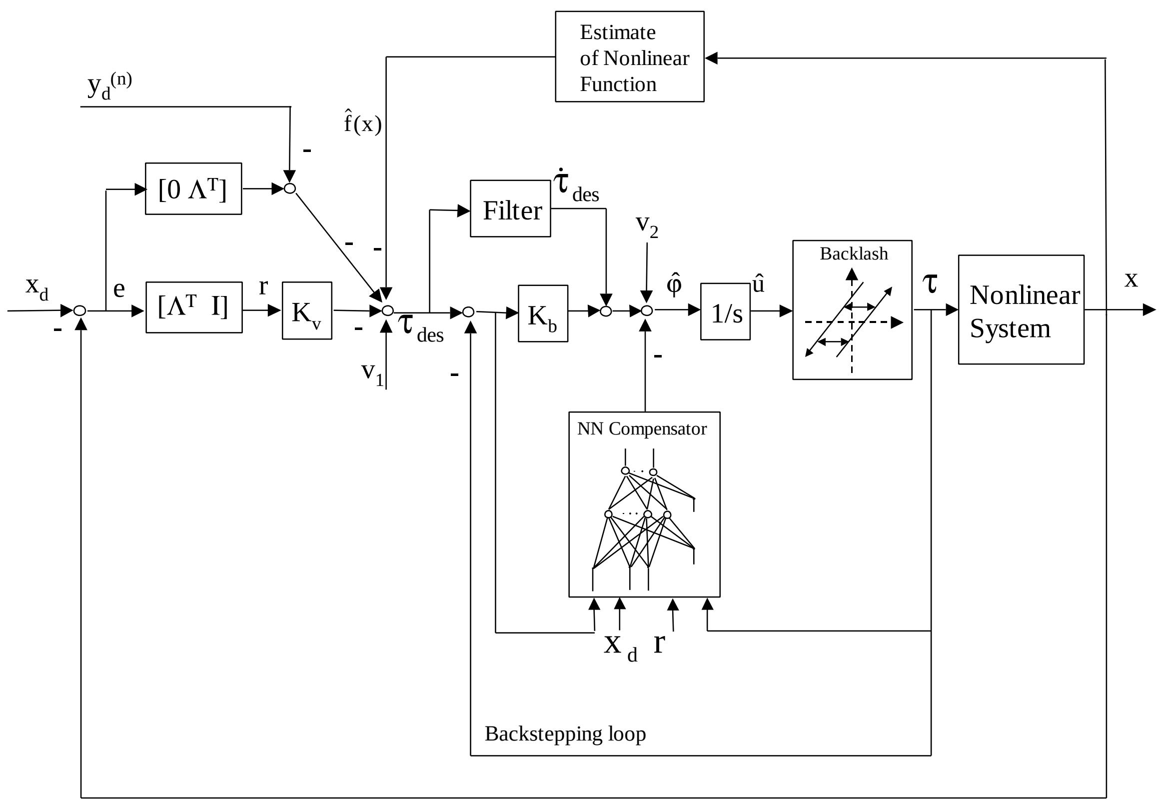

Figure 4.1 NN Backlash Compensator

Figure 4.1 shows the closed-loop system with NN backlash compensator. The proposed backlash compensation scheme is in accordance with the backlash inverse decomposition shown in Figure 3.3, i.e. the exact backlash inverse consists of a direct feed term plus the error term in Figure 3.3b which is estimated by NN.

Using the proposed controller (4.27), the error dynamics (4.18) can be written as

$$\begin{aligned}\dot{\tilde{\tau}} &= \dot{\tau}_{des} - \hat{\varphi} - \tilde{B}\left(\tau, \hat{u}, \dot{\hat{u}}\right) \\ &= -K_b\tilde{\tau} + \hat{W}^T\sigma\left(\hat{V}^T x_{nn}\right) - v_2 - W^T\sigma\left(V^T x_{nn}\right) - \varepsilon\end{aligned} \quad . \tag{4.28}$$

One can use the Taylor series expansion in order to overcome the strong restriction of linearity in the tunable parameters. Note that the first-layer weights V appear in nonlinear fashion. Applying the method developed in [16], [17] one obtains the error dynamics

$$\dot{\tilde{\tau}} = -K_b\tilde{\tau} - \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T x_{nn}\right) - \hat{W}^T\hat{\sigma}'\tilde{V}^T x_{nn} + w - v_2 \ , \tag{4.29}$$

where the disturbance term is given by

$$w = -\tilde{W}^T\hat{\sigma}' V^T x_{nn} - W^T O\left(\tilde{V}^T x_{nn}\right)^2 - \varepsilon \ , \tag{4.30}$$

with $O\left(\tilde{V}^T x_{nn}\right)^2$ denoting the higher order terms in Taylor series expansion. Assuming that the approximation property of the neural network holds, the norm of the disturbance term can be bounded from above as [16], [17]

$$\|w\| \le V_M\|\tilde{W}\|_F\|x_{nn}\| + c_1 + c_2\|\tilde{V}\|_F\|x_{nn}\| + \varepsilon_N \ , \tag{4.31}$$

where c_1 and c_2 are positive computable constants. The NN input is bounded by

$$\|x_{nn}\| \le c_3 + \|r\| + X_d + \|\tilde{\tau}\| + \|\tilde{Z}\|_F \ . \tag{4.32}$$

Combining the inequalities (4.31) and (4.32) one has

$$\|w\| \le \left(V_M\|\tilde{W}\|_F + c_2\|\tilde{V}\|_F\right)\left(c_3 + \|r\| + X_d + \|\tilde{\tau}\| + \|\tilde{Z}\|_F\right) + c_1 + \varepsilon_N \ , \tag{4.33}$$

$$\|w\| \le C_0 + C_1\|\tilde{Z}\|_F + C_2\|\tilde{Z}\|_F\|r\| + C_3\|\tilde{Z}\|_F\|\tilde{\tau}\| + C_4\|\tilde{Z}\|_F^2 \ , \tag{4.34}$$

where C_i are computable positive constants.

The next theorem shows how to tune the neural network weights so the tracking errors r(t) and $\tilde{\tau}(t)$ achieve small values while the NN weights $\hat{V}$, $\hat{W}$ are close to V, W, i.e. the weight estimation errors defined by (4.25) are bounded.

<u>Theorem 2</u> (Control law for backstepping loop).
Let assumptions 1, 2 hold. Let the desired trajectories be bounded. Select the control input as (4.27). Choose the robustifying signal v_2 as

$$v_2 = K_{Z_1}\left(\|\hat{Z}\|_F + Z_M\right)\left(\tilde{\tau} + \|r\|\frac{\tilde{\tau}}{\|\tilde{\tau}\|}\right) + K_{Z_2}\|r\|\frac{\tilde{\tau}}{\|\tilde{\tau}\|} + K_{Z_3}\left(\|\hat{Z}\|_F + Z_M\right)^2\frac{\tilde{\tau}}{\|\tilde{\tau}\|} \quad , \qquad (4.35)$$

where $K_{Z_1} > \max(C_2, C_3)$, $K_{Z_2} > 1$ and $K_{Z_3} > C_4$. Let the estimated NN weights be provided by the NN tuning algorithm

$$\dot{\hat{V}} = -Tx_{nn}\tilde{\tau}^T\hat{W}^T\hat{\sigma}' - kT\|\tilde{\tau}\|\hat{V} \quad , \qquad (4.36)$$

$$\dot{\hat{W}} = -S\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T x_{nn}\right)\tilde{\tau}^T - kS\|\tilde{\tau}\|\hat{W} \quad , \qquad (4.37)$$

with any constant matrices $S=S^T>0$, $T=T^T>0$, and $k>0$ small scalar design parameter. Then the filtered tracking error r(t), error $\tau(t)$ and NN weight estimates $\hat{V}$, $\hat{W}$ are UUB, with bounds given by (4.51), (4.52). Moreover, the error $\tau(t)$ can be made arbitrarily small by increasing the gain K_b.

<u>Proof</u>: Select the Lyapunov function candidate

$$L = L_1 + \frac{1}{2}\tilde{\tau}^T\tilde{\tau} + \frac{1}{2}\mathrm{tr}\left(\tilde{W}^T S^{-1}\tilde{W}\right) + \frac{1}{2}\mathrm{tr}\left(\tilde{V}^T T^{-1}\tilde{V}\right) \quad , \qquad (4.38)$$

which weights both errors r(t) and $\tau(t)$, and NN weights estimation errors. Taking derivative

$$\dot{L} = \dot{L}_1 + \tilde{\tau}^T\dot{\tilde{\tau}} + \mathrm{tr}\left(\tilde{W}^T S^{-1}\dot{\tilde{W}}\right) + \mathrm{tr}\left(\tilde{V}^T T^{-1}\dot{\tilde{V}}\right) \quad , \qquad (4.39)$$

and using (4.6), (4.29) one has

$$\begin{aligned}\dot{L} = {} & r^T\left(f(x) + Y_d + \tau_d + \tau\right) \\ & + \tilde{\tau}^T\left(-K_b\tilde{\tau} - \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T x_{nn}\right) - \hat{W}^T\hat{\sigma}'\tilde{V}^T x_{nn} + w - v_2\right) \\ & + \mathrm{tr}\left(\tilde{W}^T S^{-1}\dot{\tilde{W}}\right) + \mathrm{tr}\left(\tilde{V}^T T^{-1}\dot{\tilde{V}}\right)\end{aligned} \quad , \qquad (4.40)$$

$$\begin{aligned}\dot{L} &= r^T\left(f(x) + Y_d + \tau_d + \tau_{des}\right) - r^T\tilde{\tau} \\ &+ \tilde{\tau}^T\left(-K_b\tilde{\tau} - \tilde{W}^T\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T x_{nn}\right) - \hat{W}^T\hat{\sigma}'\tilde{V}^T x_{nn} + w - v_2\right). \\ &+ tr\left(\tilde{W}^T S^{-1}\dot{\tilde{W}}\right) + tr\left(\tilde{V}^T T^{-1}\dot{\tilde{V}}\right)\end{aligned} \tag{4.41}$$

$$\begin{aligned}\dot{L} &= r^T\left(f(x) + Y_d + \tau_d + \tau_{des}\right) - r^T\tilde{\tau} \\ &+ \tilde{\tau}^T\left(-K_b\tilde{\tau} + w - v_2\right) \\ &+ tr\left[\tilde{W}^T\left(S^{-1}\dot{\tilde{W}} - \left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T x_{nn}\right)\tilde{\tau}^T\right)\right] + tr\left[\tilde{V}^T\left(T^{-1}\dot{\tilde{V}} - x_{nn}\tilde{\tau}^T\hat{W}^T\hat{\sigma}'\right)\right]\end{aligned} . \tag{4.42}$$

Applying (4.9) and tuning rules yields

$$\begin{aligned}\dot{L} &= r^T\left(\tilde{f}(x) + \tau_d - K_V r + v_1\right) - r^T\tilde{\tau} + \tilde{\tau}^T\left(-K_b\tilde{\tau} + w - v_2\right) \\ &+ k\|\tilde{\tau}\| tr\left[\tilde{W}^T\left(W - \tilde{W}\right)\right] + k\|\tilde{\tau}\| tr\left[\tilde{V}^T\left(V - \tilde{V}\right)\right]\end{aligned} . \tag{4.43}$$

Using the same inequality as for (4.15), expression (4.43) can be bounded as

$$\begin{aligned}\dot{L} &\le -K_{V\min}\|r\|^2 - \|r\|(f_M + \tau_M) + \|r\|\left\|\tilde{f} + \tau_d\right\| \\ &+ k\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F\left(Z_M - \left\|\tilde{Z}\right\|_F\right) - K_{b\min}\|\tilde{\tau}\|^2 - r^T\tilde{\tau} + \tilde{\tau}^T w \\ &- \tilde{\tau}^T K_{Z_1}\left(\left\|\hat{Z}\right\|_F + Z_M\right)\left(\tilde{\tau} + \|r\|\frac{\tilde{\tau}}{\|\tilde{\tau}\|}\right) - \tilde{\tau}^T K_{Z_2}\|r\|\frac{\tilde{\tau}}{\|\tilde{\tau}\|} - \tilde{\tau}^T K_{Z_3}\left(\left\|\hat{Z}\right\|_F + Z_M\right)^2\frac{\tilde{\tau}}{\|\tilde{\tau}\|}\end{aligned} . \tag{4.44}$$

Including (4.34) and applying some norm properties, one has

$$\begin{aligned}\dot{L} &\le -K_{V\min}\|r\|^2 - \|r\|(f_M + \tau_M) + \|r\|\left\|\tilde{f} + \tau_d\right\| \\ &+ k\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F\left(Z_M - \left\|\tilde{Z}\right\|_F\right) - K_{b\min}\|\tilde{\tau}\|^2 + \|\tilde{\tau}\|\|r\| \\ &+ C_0\|\tilde{\tau}\| + C_1\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F + C_2\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F\|r\| + C_3\left\|\tilde{Z}\right\|_F\|\tilde{\tau}\|^2 + C_4\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F^2 \\ &- K_{Z_1}\|\tilde{\tau}\|^2\left\|\tilde{Z}\right\|_F - K_{Z_1}\|r\|\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F - K_{Z_2}\|\tilde{\tau}\|\|r\| - K_{Z_3}\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F^2\end{aligned} , \tag{4.45}$$

$$\begin{aligned}\dot{L} &\le -K_{V\min}\|r\|^2 + kZ_M\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F - k\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F^2 - K_{b\min}\|\tilde{\tau}\|^2 + \|\tilde{\tau}\|\|r\| \\ &+ C_0\|\tilde{\tau}\| + C_1\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F + C_2\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F\|r\| + C_3\left\|\tilde{Z}\right\|_F\|\tilde{\tau}\|^2 + C_4\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F^2 \\ &- K_{Z_1}\|\tilde{\tau}\|^2\left\|\tilde{Z}\right\|_F - K_{Z_1}\|r\|\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F - K_{Z_2}\|\tilde{\tau}\|\|r\| - K_{Z_3}\|\tilde{\tau}\|\left\|\tilde{Z}\right\|_F^2\end{aligned} . \tag{4.46}$$

Taking $K_{Z_2} > 1$ yields

$$\begin{aligned}\dot{L} \le & -K_{V\min}\|r\|^2 + kZ_M\|\tilde{\tau}\|\|\tilde{Z}\|_F - k\|\tilde{\tau}\|\|\tilde{Z}\|_F^2 - K_{b\min}\|\tilde{\tau}\|^2 \\ & + C_0\|\tilde{\tau}\| + C_1\|\tilde{\tau}\|\|\tilde{Z}\|_F + C_2\|\tilde{\tau}\|\|\tilde{Z}\|_F\|r\| + C_3\|\tilde{Z}\|_F\|\tilde{\tau}\|^2 + C_4\|\tilde{\tau}\|\|\tilde{Z}\|_F^2 \ . \\ & - K_{Z_1}\|\tilde{\tau}\|^2\|\tilde{Z}\|_F - K_{Z_1}\|r\|\|\tilde{\tau}\|\|\tilde{Z}\|_F - K_{Z_3}\|\tilde{\tau}\|\|\tilde{Z}\|_F^2\end{aligned} \tag{4.47}$$

Choosing $K_{Z_1} > \max(C_2, C_3)$ and $K_{Z_3} > C_4$ one has

$$\begin{aligned}\dot{L} \le & -K_{V\min}\|r\|^2 + kZ_M\|\tilde{\tau}\|\|\tilde{Z}\|_F - k\|\tilde{\tau}\|\|\tilde{Z}\|_F^2 - K_{b\min}\|\tilde{\tau}\|^2 \\ & + C_0\|\tilde{\tau}\| + C_1\|\tilde{\tau}\|\|\tilde{Z}\|_F\end{aligned} \tag{4.48}$$

$$\dot{L} \le -K_{V\min}\|r\|^2 - \|\tilde{\tau}\|\left(K_{b\min}\|\tilde{\tau}\| + k\|\tilde{Z}\|_F^2 - (kZ_M + C_1)\|\tilde{Z}\|_F - C_0\right). \tag{4.49}$$

Completing the squares yields

$$\dot{L} \le -K_{V\min}\|r\|^2 - \|\tilde{\tau}\|\left[K_{b\min}\|\tilde{\tau}\| + k\left(\|\tilde{Z}\|_F - \left(\frac{Z_M k + C_1}{2k}\right)\right)^2 - k\left(\frac{Z_M k + C_1}{2k}\right)^2 - C_0\right]. \tag{4.50}$$

Thus, the $\dot{L}$ is negative as long as

$$\|\tilde{\tau}\| > \frac{k\left(\frac{Z_M k + C_1}{2k}\right)^2 + C_0}{K_{b\min}} \tag{4.51}$$

$$\|\tilde{Z}\|_F > \frac{Z_M k + C_1}{2k} + \sqrt{\left(\frac{Z_M k + C_1}{2k}\right)^2 + \frac{C_0}{k}} \tag{4.52}$$

The first terms of (4.36), (4.37) are modified versions of the standard backpropagation algorithm. The k terms correspond to the *e*-modification [21], to guarantee bounded parameter estimates. Note that robustifying term consists of three terms. The first and third terms are specifically designed to ensure the stability of the overall system in the presence of disturbance term (4.34). The second term is given to ensure the stability due to the error $\tilde{\tau} = \tau_{des} - \tau$ in the backstepping design.

The right-hand side of (4.51) can be taken as a practical bound on the error in the sense that $\tilde{\tau}(t)$ will never stay far above it. Note that the stability radius may be decreased any amount by increasing the gain K_b. It is noted that PD control without backlash compensation requires much higher gain in order to achieve the similar performance– that is, eliminating the NN feedforward compensator will result in degraded performance. Moreover, it is difficult to guarantee the stability of such highly nonlinear system using only PD. Using the NN backlash compensation, stability of the system is proven, and the tracking error can be kept arbitrarily small by increasing the gain K_b. The NN weight errors are fundamentally bounded in terms of V_M, W_M.

Due to the form of the feedforward compensator, which has a unity feedforward path plus a NN parallel path, it is straightforward to initialize the NN weights. The initial weights V are selected randomly, while the initial weights W are set to zero. Then the PD loop with unity gain feedforward path holds the system stable until the NN begins to learn.

5 SIMULATION OF NN BACKLASH COMPENSATOR

To illustrate the performance of the NN backlash compensator, we consider the nonlinear system

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{1}{T}x_2 + ma\,x_2^2\sin(x_1) + mga\cos(x_1) + \tau \end{aligned} \tag{5.1}$$

which represents a mechanical motion of robot-like system with one link. The motor time constant is *T*, *m* is a net effective load mass, *a* a length, and *g* the gravitational constant. We selected *T*=1 s; *m*=1 kg; *a*=2.5 m. The input t is passed through the additional backlash nonlinearity given by Equation (3.1). The parameters of the backlash are d_+=10, d_-=-10.5, m=1.

The NN weight tuning parameters are chosen as $S = 10 I_{11}$, $T = 10 I_6$, k=0.001, where I_N is NxN identity matrix. The robustifying signal gains are K_{z1}=10, K_{z2}=10, K_{z3}=10. The controller parameters are chosen as Λ=10, K_v=10, K_b=50.

The NN has L=10 hidden-layer nodes with sigmoidal activation functions. The first-layer weights V are initialized randomly [25]. They are uniformly randomly distributed between -1 and 1. Second-layer weights W are initialized at zero. Note that this weight initialization will not affect system stability since the weights W are initialized at zero, and therefore there is initially no input to the system except for the PD loop. Filter that generates the signal $\dot{\tau}_{des}$ is implemented as $\frac{s}{s+100}$.

Figure 5.1 and Figure 5.2 show the tracking performance of the closed-loop system without the NN backlash compensation. It can be seen that backlash degrades the system performance, since it causes the lost of information about the control signal u(t), whenever u(t) change its direction. Desired trajectory which is shown in the figures is sin(t).

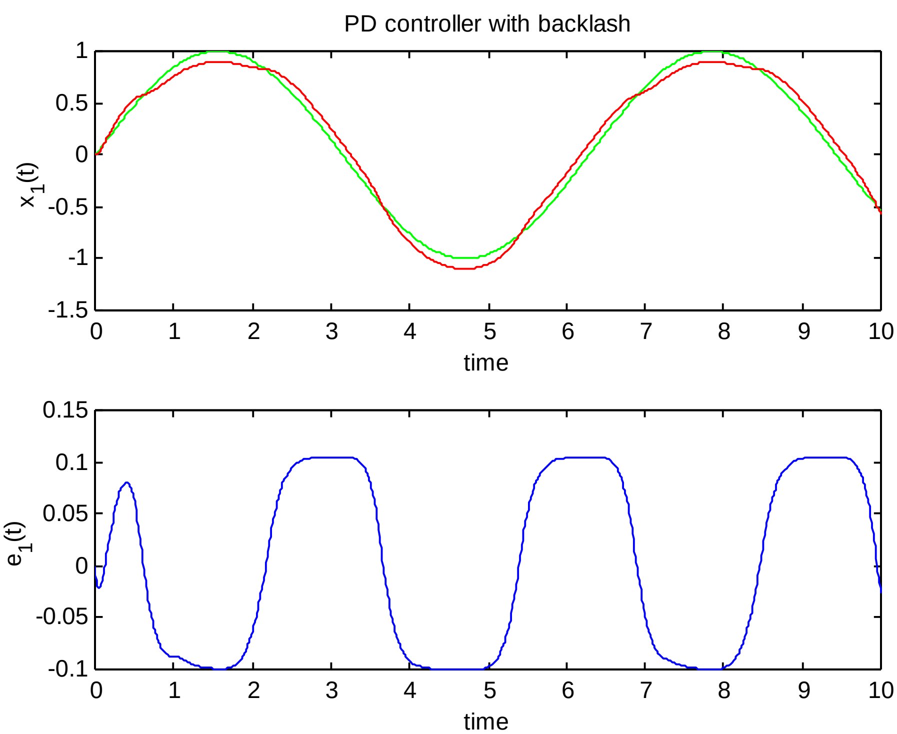

Figure 5.1 State $x_1(t)$ and tracking error $e_1(t)$ without backlash compensation.

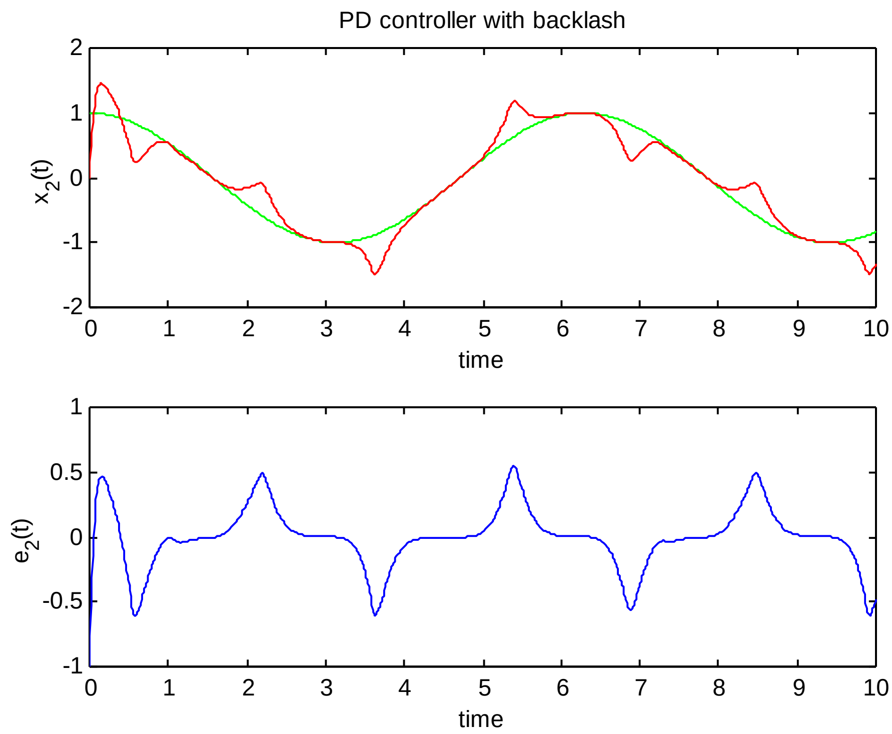

Figure 5.2 State $x_2(t)$ and tracking error $e_2(t)$ without backlash compensation.

Applying the NN backlash compensator greatly reduces the tracking error. Figure 5.3 and Figure 5.4 show tracking errors when NN compensator is included. One can see that after 0.5s neural network adjusts its weights on-line, such that the backlash effect is mostly reduced. Figure 5.5 show the control signal u(t) in both cases, when NN is applied and without NN compensator. It is interesting to note that when the NN is used, the control signal u(t) is in the “envelope” of the control signal u(t) without the NN. The control u(t) has a superimposed high-frequency component that is very similar to that injected using dithering techniques. Therefore, one could consider adaptive NN backlash compensator as an *adaptive dithering technique* [25], where the NN inserts dithering signal.

Some of the neural network weights are shown in the Figure 5.6. One can see that the NN weights W vary through time, truing to compensate for the system nonlinearity, while NN weights V which represents the position of the sigmoid functions vary through time very slowly.

From this simulation it is clear that the proposed NN backlash compensator is an efficient way to compensate for backlash nonlinearities of all kind, without any restrictive assumptions on the backlash model itself.

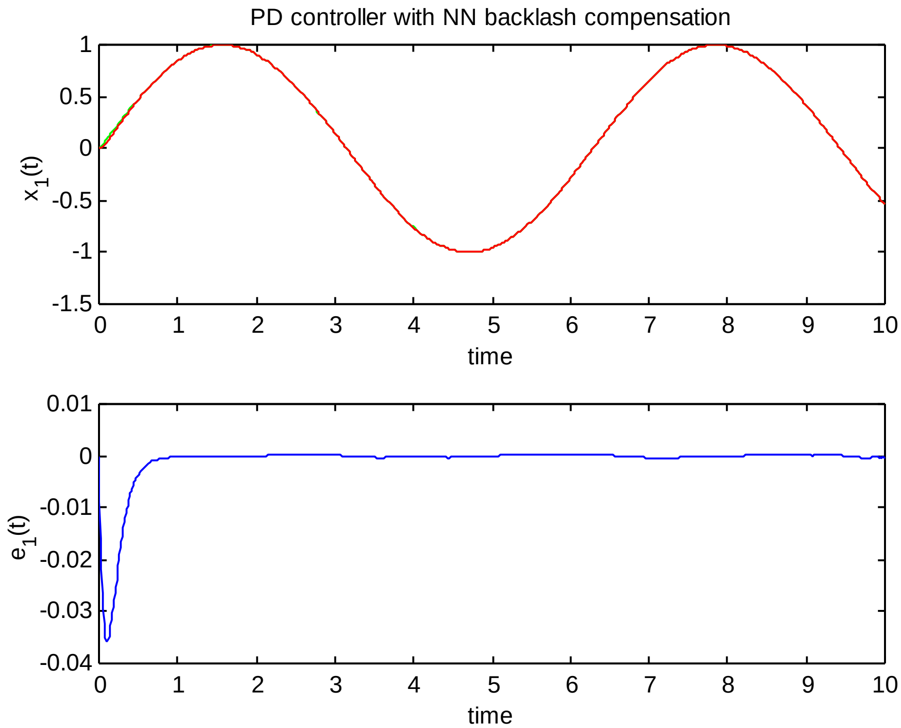

Figure 5.3 State $x_1(t)$ and tracking error $e_1(t)$ with NN backlash compensation.

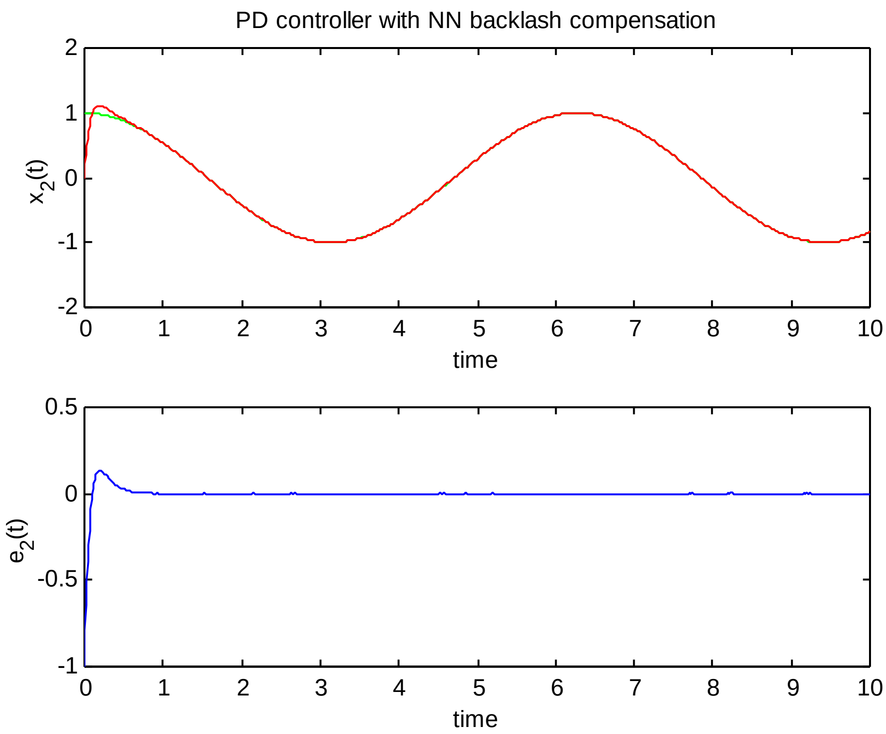

Figure 5.4 State $x_2(t)$ and tracking error $e_2(t)$ with NN backlash compensation.

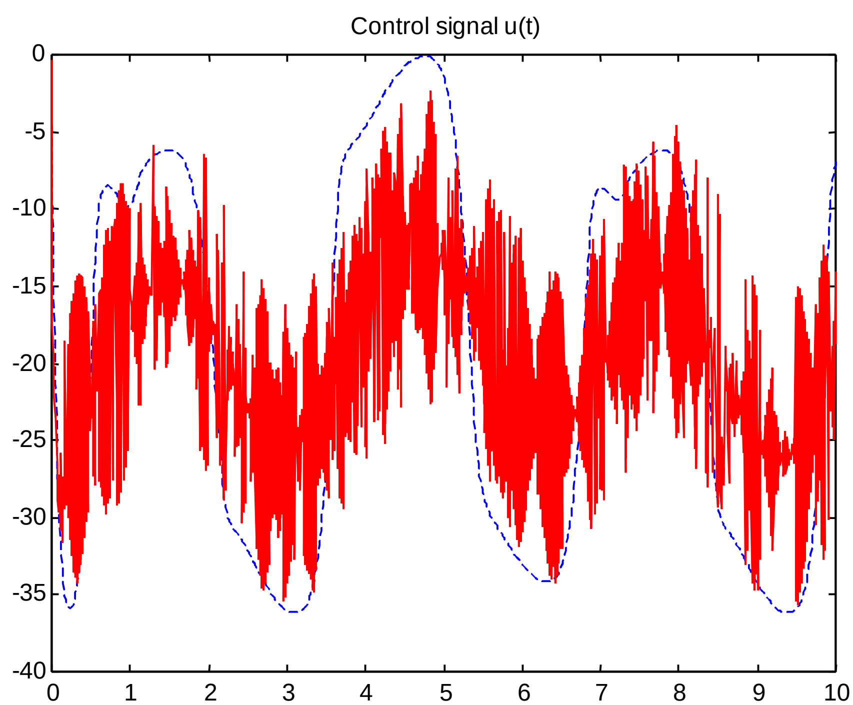

Figure 5.5 Control signal u(t): without backlash compensation (dash), and with NN backlash compensator (full).

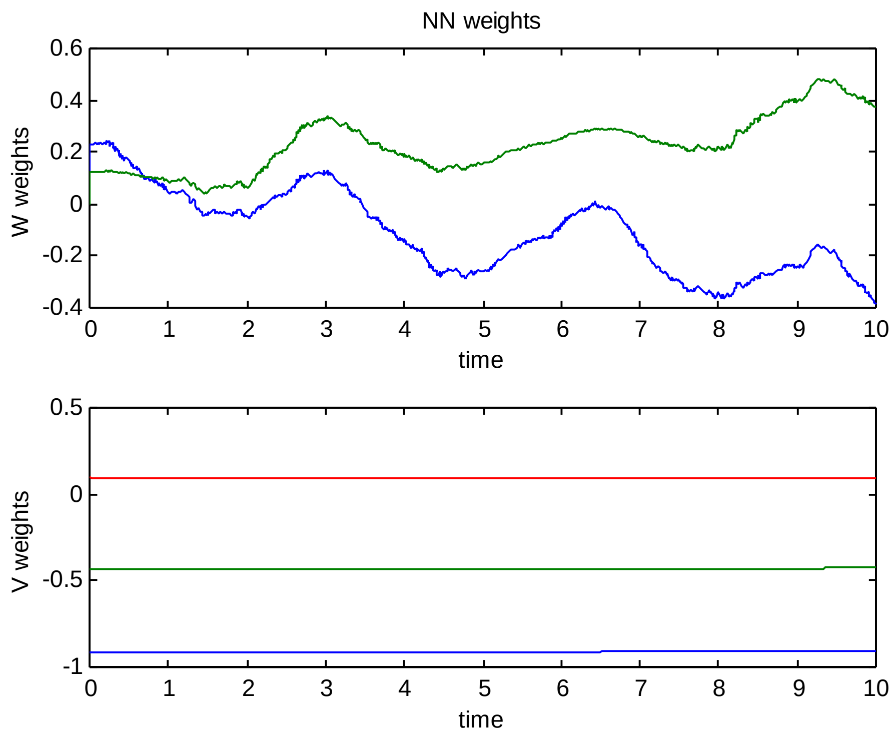

Figure 5.6 Some of the neural network weights.

6 CONCLUSION

A new technique for the backlash compensation is presented. It does not require any restrictive assumptions on the backlash nonlinearity (e.g. the same slopes of the lines, etc.). The compensator scheme has dynamic inversion structure, with the NN in the feedforward path. The NN controller does not require preliminary off-line training. Rigorous stability proofs are given using Lyapunov theory. Simulation results show that the proposed compensation scheme is efficient way of improving the tracking performance of nonlinear systems with backlash.

REFERENCES

[1] R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Trans. Info. Theory*, vol. 39, no. 3, pp. 930-945, May 1993.

[2] R. G. Bartle, *The Elements of Real Analysis*, Wiley, New York, 1964.

[3] F.-C. Chen, and H. K. Khalil, "Adaptive control of nonlinear systems using neural networks," *Int. J. Contr.*, vol. 55, no. 6, pp. 1299-1317, 1992.

[4] M. J. Corless and G. Leitmann, "Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamic systems," *IEEE Trans. Automat. Contr.*, vol. 26, no. 5, pp. 850-861, 1982.

[5] J. J. Craig, *Adaptive Control of Robot Manipulators*, Addison-Wesley, Reading, MA, 1988.

[6] Cybenko, "Approximation by superpositions of a sigmoidal function," *Math. Contr. Signals, Syst.*, vol. 2, no. 4, pp. 303-314, 1989.

[7] C. A. Desoer and S. M. Shahruz, "Stability of dithered nonlinear systems with backlash or hysteresis," *Int. J. Contr.*, vol. 43, no. 4, pp. 1045-1060, 1986.

[8] B. Igelnik and Y. H. Pao, "Stochastic Choice of Basis Functions in Adaptive Function Approximation and the Functional-Link Net," *IEEE Trans. Neural Networks*, vol. 6, no. 6, pp. 1320-1329, Nov. 1995.

[9] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*, John Wiley & Sons, New York, NY, 1995.

[10] B. S. Kim and A. J. Calise, "Nonlinear flight control using neural networks," *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 1, Jan.-Feb. 1997.

[11] J.-H. Kim, J.-H. Park, S.-W. Lee, and E. K. P. Chong, "A two-layered fuzzy logic controller for systems with deadzones," *IEEE Trans. Industrial Electron.*, vol. 41, no. 2, pp. 155-162, Apr. 1994.

[12] C. Kwan, F. L. Lewis, and D. M. Dawson, "Robust neural-network control of rigid-link electrically driven robots," *IEEE Trans. Neural Networks*, vol. 9, no. 4, pp. 581-588, July 1998.

[13] S.-W. Lee and J.-H. Kim, “Control of systems with deadzones using neural-network based learning control,” *Proc. IEEE Int. Conf. Neural Networks*, June 1994, pp. 2535-2538.

[14] J. Leitner, A. Calise and J. V. R. Prasad, “Analysis of adaptive neural networks for helicopter flight control,” *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 5, pp. 972-979, Sep.-Oct. 1997.

[15] F. L. Lewis, C. T. Abdallah, and D. M. Dawson, *Control of Robot Manipulators*, Macmillan, New York, 1993.

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

[17] F. L. Lewis, A. Yesildirek, and K. Liu, “Multilayer neural-net robot controller with guaranteed tracking performance,” *IEEE Trans. Neural Networks*, vol. 7, no. 2, pp. 1-11, Mar. 1996.

[18] F. L. Lewis, K. Liu, R. R. Selmic, and Li-Xin Wang, “Adaptive fuzzy logic compensation of actuator deadzones,” *J. Robot. Sys.*, vol. 14, no. 6, pp. 501-511, 1997.

[19] W. Li and X. Cheng, “Adaptive high-precision control of positioning tables-- theory and experiment,” *IEEE Trans. Control Syst. Technol.*, vol. 2, no. 3, pp. 265-270, Sep. 1994.

[20] M. B. McFarland and A. J. Calise, “Multilayer neural networks and adaptive nonlinear control of agile anti-air missiles,” *Preprint*, *1999.*

[21] K. S. Narendra and A. M. Annaswamy, “A new adaptive law for robust adaptation without persistent excitation,” *IEEE Trans. Automat. Control*, vol. 32, no. 2, pp. 134-145, Feb. 1987.

[22] K. S. Narendra, “Adaptive Control Using Neural Networks,” *Neural Networks for Control*, pp. 115-142. ed. W. T. Miller, R. S. Sutton, P. J. Werbos, Cambridge: MIT Press, 1991.

[23] K. S. Narendra and K. Parthasarathy, “Identification and control of dynamical systems using neural networks,” *IEEE Trans. Neural Networks*, vol. 1, pp. 4-27, March 1990.

[24] M. M. Polycarpou, "Stable adaptive neural control scheme for nonlinear systems," *IEEE Trans. Automat. Contr.*, vol. 41, no. 3, pp. 447-451, March 1996.

[25] Recker, P. V. Kokotovic, D. Rhode, and J. Winkelman, "Adaptive nonlinear control of systems containing a dead-zone," *Proc. IEEE Conf. Decis. Control*, 1991, pp. 2111-2115.

[26] G. A. Rovithakis and M. A. Christodoulou, "Adaptive control of unknown plants using dynamical neural networks," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 24, no. 3, pp. 400-412, 1994.

[27] N. Sadegh, "A perceptron network for functional identification and control of nonlinear systems," *IEEE Trans. Neural Networks*, vol. 4, no. 6, pp. 982-988, Nov. 1993.

[28] R. M. Sanner and J. -J. E. Slotine, "Stable adaptive control and recursive identification using radial gaussian networks," *Proc. IEEE Conf. Decis. Control*, Brighton, 1991.

[29] R. R. Selmic and F. L. Lewis, "Neural network approximation of piecewise continuous functions: application to friction compensation," *Proc. IEEE Int. Symp. Intell. Contr.*, Istanbul, July 1997.

[30] R. R. Selmic and F. L. Lewis, "Deadzone compensation in motion control systems using neural networks," *submitted to IEEE Trans. Automat. Control.*

[31] R. R. Selmic and F. L. Lewis, "Deadzone compensation in nonlinear systems using neural networks," *Proc. IEEE Conf. Decis. Control*, Tampa, FL, 1998

[32] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice-Hall, New Jersey, 1991.

[33] Y. D. Song, T. L. Mitchell, and H. Y. Lai, "Control of a class of nonlinear uncertain systems via compensated inverse dynamics approach," *IEEE Trans. Automat. Contr.*, vol. 39, no. 9, pp. 1866-1871, September 1994.

[34] G. Tao and P. V. Kokotovic, "Adaptive control of plants with unknown dead-zones," *IEEE Trans. Automat. Control*, vol. 39, no. 1, pp. 59-68, Jan. 1994.

[35] G. Tao and P. V. Kokotovic, “Continuous-time adaptive control of systems with unknown backlash,” *IEEE Trans. Automat. Control*, vol. 40, no. 6, pp. 1083-1087, June 1995.

[36] G. Tao and P. V. Kokotovic, *Adaptive Control of Systems With Actuator and Sensor Nonlinearities*, John Wiley & Sons, New York, 1996.

[37] A. Tzes, P.-Y. Peng and C.-C. Houng, “Neural network control for DC motor micromaneuvering,” *IEEE Trans. Ind. Electron.*, vol. 42, no.5, pp. 516-523, October 1995.

[38] L. Yan and C. J. Li, “Robot learning control based on recurrent neural network inverse model,” *J. Robot. Sys.*, vol. 14, no. 3, pp. 199-211, 1997.