BACKLASH COMPENSATION IN NONLINEAR SYSTEMS USING DYNAMIC INVERSION BY NEURAL NETWORKS

Rastko R. Selmic and Frank L. Lewis

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.

KeyWords: Neurocontrol, neural networks, backlash compensation, actuator nonlinearity control, dynamic inversion.

I. 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 [40]. Backlash compensation is considered in [39]. Compensation for nonsymmetric deadzones is considered in [38] for linear systems, in [27] for nonlinear systems in Brunovksy form with known nonlinear functions. For the dynamic system in Lagrangian form, deadzone compensation using NN is given in [32, 33]. 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 [42]. A compensated inverse dynamics approach using adaptive and robust control techniques is presented in [35].

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 Brunovksy form. Simulation results show that NN backlash compensator can significantly reduce degrading effect of backlash nonlinearity.

II. BACKGROUND

Let $S$ be a compact simply connected set of $\mathbb{R}^n$. With map $f: S \rightarrow \mathbb{R}^m$, 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 $\|\cdot\|_p$ it 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 $\|f\|$. 

Manuscript received October 8, 1999; revised February 3, 1999; accepted May 10, 2000.

Rastko R. Selmic and Frank L. Lewis are with Automation and Robotics Research Institute, The University of Texas at Arlington, 7300 Jack Newell Blvd., South Fort Worth, Texas 76118-7115, U.S.A.
sup \( f(x) \mid f : S \to \mathbb{R}^m \). \hspace{1cm} (2.1)

Given \( A = [a_i], B \in \mathbb{R}^{m \times n} \) the Frobenius norm is defined by
\[
\|A\|_F = \text{tr}(A^T A) = \sum_{i,j} a_{ij}^2, \hspace{1cm} (2.2)
\]
with \( \text{tr}(\cdot) \) the trace. The associated inner product is \( \langle A, B \rangle_F = \text{tr}(A^T B) \). The Frobenius norm is compatible with the 2-norm so that \( \|Ax\|_2 \leq \|A\|_F \|x\|_2 \).

When \( x(t) \in \mathbb{R}^n \) is a function of time, we use the standard \( L_{\infty} \) norms. It is said that \( x(t) \) is bounded if its \( L_{\infty} \) norm is bounded. Matrix \( A(t) \in \mathbb{R}^{m \times n} \) is bounded if its induced matrix \( \infty \)-norm is bounded.

Consider the nonlinear system
\[
\dot{x} = g(x, u, t), \quad y = h(x, t) \hspace{1cm} (2.3)
\]
with state \( x(t) \in \mathbb{R}^n \). The equilibrium point \( x_e \) is said to be uniformly ultimately bounded (UUB) if there exists a compact set \( S \subset \mathbb{R}^n \) so that for all \( x_0 \in S \) there exists an \( \varepsilon > 0 \) and a number \( T(\varepsilon, x_0) \) such that \( \|x(t) - x_e\| \leq \varepsilon \) for all \( t \geq t_0 + T \). That is, after a transition period \( T \), the state \( x(t) \) remains within the ball of radius \( \varepsilon \) around \( x_e \).

2.1 Background on Neural Networks

NN have been used extensively in feedback control systems. Most applications are \textit{ad hoc} with no demonstrations of stability. The stability proofs that do exist rely almost invariably on the universal approximation property for NN \([3,17,28,29,30]\). NN approximation of discontinuous functions with application to friction and deadzone compensation is given in \([31,32,33]\).

The two-layer NN in Fig. 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 \( \mathbb{R}^n \) into output space \( \mathbb{R}^m \).

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
\[
y = \rho \left( \sum_{i=1}^{L} w_i \sigma \left( \sum_{j=1}^{L} v_{ij} x_j + v_{i0} \right) + w_{i0} \right), \hspace{1cm} (2.4)
\]
where \( \sigma(\cdot) \)s 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 \( v_{ij} \) and the second-layer interconnection weights by \( w_{i0} \). The threshold offsets are denoted by \( v_{i0}, w_{i0} \).

By collecting all the NN weights \( v_{ij}, w_{i0} \) into matrices \( V^T, W^T \), the NN equation with linear output activation function \( \rho(\cdot) \) may be written in terms of vectors as
\[
y = W^T \sigma(V^T x). \hspace{1cm} (2.5)
\]
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. \( x = [1 \ x_1 \ x_2 \ldots \ x_n]^T \)). In this equation, to represent (2.4) one has sufficient generality if \( \sigma(\cdot) \) is taken as a diagonal function from \( \mathbb{R}^l \) to \( \mathbb{R}^l \), that is \( \sigma(z) = \text{diag}\{\sigma(z_i)\} \) for a vector \( z = [z_1 \ z_2 \ldots z_L]^T \in \mathbb{R}^L \). For notational convenience we define the matrix of all the weights as
\[
Z = \begin{bmatrix} W \\ V \end{bmatrix}. \hspace{1cm} (2.6)
\]

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
\[
\sigma(x) = \frac{1}{1 + e^{-x}}. \hspace{1cm} (2.7)
\]

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 \( \mathbb{R}^n \), and any \( \varepsilon_\delta > 0 \), one has
\[
f(x) = W^T \sigma(V^T x) + \varepsilon(x), \hspace{1cm} (2.8)
\]
where the \( \varepsilon(x) \) is the NN approximation error, and \( \|\varepsilon(x)\| \leq \varepsilon_\delta \) 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 \((1/L)^2\). The approximating weights \(V\) and \(W\) are ideal target weights, and it is assumed that they are bounded such that \(\|V\|_F \leq V_{\text{max}}\) and \(\|W\|_F \leq W_{\text{max}}\), or \(\|Z\|_F \leq Z_{\text{max}}\).

**III. 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 Fig. 3.1, and a mathematical model is given by (3.1) [40].

\[
\tau = B(\tau, u, \dot{u}) = \begin{cases} 
mu, & \text{if } \dot{u} > 0 \text{ and } \tau = mu - md \\
0, & \text{otherwise}
\end{cases}
\]

(3.1)

One can see that backlash is a first-order velocity-driven dynamic system, with inputs \(u\) and \(\dot{u}\), and state \(\tau\).

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 Fig. 3.2.

The dynamics of the backlash compensator is given by

\[
\dot{u} = B_{\text{inv}}(u, w, \dot{w})
\]

(3.2)

Note that the backlash inverse is a dynamic system. It requires dynamic compensation techniques or dynamic inversion in the feedforward path.

**IV. NN CONTROLLER WITH BACKLASH COMPENSATION**

The NN backlash compensator is designed using the backstepping technique originally developed by Krustic 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

\[
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_3 \\
&\vdots \\
\dot{x}_n &= f(x) + \tau_d + \tau \\
y &= x_1
\end{align*}
\]

(4.1)
where the output is $y(t)$, the state is $x = [x_1, x_2, \ldots, x_n]^T$, $\tau_d$ 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 $|\tau_d| \leq \tau_{d,\text{p}}$ with $\tau_d(t)$ 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, $\hat{f}(x) = f(x) - \hat{f}(x)$, satisfies

$$|\hat{f}(x)| \leq f_{\text{ub}}(x), \quad (4.2)$$

for some known bounding function $f_{\text{ub}}(x)$.

This is not unreasonable [4,14], as in practical systems the bound $f_{\text{ub}}(x)$ 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

$$x_d(t) = [y_d, \dot{y}_d, \ldots, y_d^{(n-1)}]^T, \quad (4.3)$$

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

$$e = x - x_p, \quad (4.4)$$

and the filtered tracking error by

$$r = \begin{bmatrix} \lambda_1 & \lambda_2 & \ldots & \lambda_{n-1} & 1 \end{bmatrix} e \equiv [A^T \ 1] e, \quad (4.5)$$

with being $A$ a gain parameter vector selected so that $e(t) \to 0$ exponentially as $r(t) \to 0$. Then, (4.5) is a stable system so that $e(t)$ is bounded as long as controller guarantees 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

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

where

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

is a known function of the desired trajectory and actual states.

**Assumption 3. (Bounded desired trajectory)** The desired trajectory is bounded so that

$$|x_d(t)| \leq X_{d,\text{p}}, \quad (4.8)$$

where $X_{d,\text{p}}$ 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

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

with $\hat{f}(x)$ being 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 $K_y > 0$ 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 [4,15]. If $f(x)$ in (4.1) is unknown, it can be estimated using adaptive control techniques [5,15], 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 $\tau_{\text{des}}$ 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

$$v_1(t) = - \left( f_{\text{ub}}(x) + \tau_{\text{des}} \right) \frac{r}{|r|}. \quad (4.10)$$

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

**Proof.** Select the Lyapunov function candidate

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

Differentiating $L_1$ and using (4.6) yields

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

Applying the tracking control law (4.9) one has
\[
\dot{L}_1 = r^T \left[ f(x) + Y_d + \tau_d - K_v r - \dot{f}(x) - Y_d + v_1 \right].
\]

(4.13)

\[
\dot{L}_1 = r^T \left[ \ddot{f}(x) + \tau_d - K_v r + v_1 \right].
\]

(4.14)

\[
\dot{L}_1 = -r^T K_v r + r^T \left[ \ddot{f}(x) + \tau_d - \left( f_m(x) + \tau_m \right) \right].
\]

(4.15)

Expression (4.15) can be bounded as
\[
L_1 \leq -K_{\text{v, max}} \left| r \right|^2 - \left| r \right| \left| f_m + \tau_m \right| + \left| r \right| \left| \dot{f} + \tau_r \right|.
\]

(4.16)

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

4.3 NN Backlash Compensation Using Dynamic Inversion

In Theorem 1, the control law is given 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 \( \tau \) 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. [10,20], 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
\[
\tau_e = \tau_{\text{des}} - \tau.
\]

(4.17)

after differentiation, one has
\[
\dot{\tau}_e = \tau_{\text{des}} - \dot{\tau} = \tau_{\text{des}} - B(\tau, u, \dot{u})
\]

(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
\[
\tau = \varphi = B(\tau, u, \dot{u})
\]

(4.19)

(4.20)

where \( \varphi(t) \) is pseudo-control input [14,20]. In the case of known backlash, the ideal backlash inverse is given by
\[
\dot{u} = B^{-1}(u, \tau, \varphi).
\]

(4.21)

Since the backlash and therefore its inverse are not known, one can only approximate the backlash inverse
\[
\hat{u} = \hat{B}^{-1}(\hat{u}, \tau, \varphi).
\]

(4.22)

The backlash dynamics can now be written as
\[
\tau = B(\tau, \hat{u}, \hat{u})
\]

(4.23)

where
\[
\varphi = \tilde{B}(\tau, \hat{u}, \hat{u}),
\]

(4.24)

and therefore its inverse is
\[
\hat{u} = \hat{B}^{-1}(\tau, \hat{u}, \varphi).
\]

(4.25)

The unknown function \( \tilde{B}(\tau, \hat{u}, \hat{u}) \), which represents the backlash inversion error, will be approximated using a neural network.

Based on the NN approximation property, the backlash inversion error can be represented as
\[
\tilde{B}(\tau, \hat{u}, \hat{u}) = W^T \sigma(V^T x_{\text{NN}}) + \varepsilon(x_{\text{NN}}),
\]

(4.26)

where \( \varepsilon(x_{\text{NN}}) \) is the NN approximation error and is bounded on a compact set. Note that \( \tilde{B}(\tau, \hat{u}, \hat{u}) \) is not smooth function, since backlash inverse itself is not smooth. The compensation technique is general enough to include additional unmodeled dynamics besides backlash at the system input. Sandberg et al [24,25], and Sontag [37] have shown that such NN can be used for approximation of piecewise continuous functions. The output \( y_{\text{NN}} \) of the NN is given by
\[
y_{\text{NN}} = \hat{W}^T \sigma(V^T x_{\text{NN}}).
\]

(4.27)

The NN input vector is chosen as \( x_{\text{NN}} = [1 \ r^T \ x_d^T \ \tau^T \ y_{\text{NN}}^T \ Z^T \ y_{\text{NN}}^T]^T \). The NN input elements are chosen based on the functional dependence of the function \( \tilde{B}(\tau, \hat{u}, \hat{u}) \). Note that the network output is also the NN input. This requires the fixed-point solution assumption, which holds for bounded sigmoidal activation functions [10,20].

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
\[
\dot{V} = V - \dot{V}, \quad \dot{W} = W - \dot{W}, \quad \dot{Z} = Z - \dot{Z},
\]
and the hidden-layer output error for a given \( x \) as
\[
\dot{\sigma} = \sigma - \dot{\sigma} = \sigma(V^T x_{nn}) - \sigma(\dot{V}^T x_{nn}).
\]

In order to design the stable closed-loop system with backlash compensation, one selects nominal backlash inverse \( \hat{u} = B^{-1}(\hat{u}, \tau, \phi) = \phi \) and pseudo-control input \( \phi \) as
\[
\phi = K_b \tau + \tau - W^T \sigma(V^T x_{nn}) + v_2,
\]
where \( v_2(t) \) is a robustifying term detailed later. The backlash inversion error will be approximated using NN.

Note that we do not use two NN as in [32] since we assume that the signal \( \tau \) is measurable. If \( \tau \) is not measurable, then the compensator structure would involve two NNs, one as backlash estimator, and the other one as compensator.

Figure 4.1 shows the closed-loop system with NN backlash compensator. The proposed backlash compensation scheme consists of a direct feed term plus the error term, which is estimated by NN.

Using the proposed controller (4.30), the error dynamics (4.18) can be written as
\[
\hat{\tau} = \tau_{des} - \phi - B(\tau, \hat{u}, \hat{u})
\]
\[
= -K_b \hat{\tau} + W^T \sigma(V^T x_{nn}) - v_2 - W^T \sigma(V^T x_{nn}) - \varepsilon.
\]

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
\[
\dot{\tau} = -K_b \hat{\tau} - W^T (\hat{\sigma} - \dot{\sigma} \cdot V^T x_{nn}) - \dot{W}^T \hat{\sigma} V^T x_{nn} + w - v_2,
\]
where the disturbance term is given by
\[
w = -W^T \sigma V^T x_{nn} - W^T O(V^T x_{nn})^2 - \varepsilon(x),
\]
with \( O(V^T x_{nn})^2 \) 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]
\[
\|w\| \leq V M \|\hat{W}\|_F x_{nn} + c_1 + c_2 \|\hat{V}\|_F x_{nn} + \varepsilon_N,
\]
where \( c_1 \) and \( c_2 \) are positive computable constants. The NN input is bounded by
\[
\|x_{nn}\| \leq c_3 + \|r\| + X_d + \|\hat{\tau}\| + c_4 \|\hat{Z}\|_F.
\]

Combining the inequalities (4.34) and (4.35) one has

Figure 4.1 NN Backlash compensator.
\[
\begin{align*}
|v| & \leq (V_m |\bar{W}|_F + c_3 |\bar{Z}|_F) + c_5 + |r| + c_6 |\tilde{x}| + c_7 + \varepsilon_v, \\
|v| & \leq C_0 + C_1 |\bar{Z}|_F + C_2 |\bar{Z}|_F |r| + C_3 |\bar{Z}|_F |\bar{Z}|_F, \\
|v| & \leq C_0 + C_1 |\bar{Z}|_F + C_2 |\bar{Z}|_F |r| + C_3 |\bar{Z}|_F |\bar{Z}|_F, \\
\end{align*}
\]

(4.36)

(4.37)

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 \(\tilde{x}(t)\) achieve small values while the NN weights \(V, W\) are close to \(V, \bar{W}\), i.e. the weight estimation errors defined by (4.28) are bounded.

**Theorem 2. (Control law for backstepping loop)** Let assumptions 1, 2 hold. Let the desired trajectories be bounded. Select the control input as (4.30). Choose the robustifying signal \(v_2\) as

\[
v_2 = K_{z_1} \left( |\bar{Z}|_F + Z_m \right) \left( \tilde{x} + |r| \tilde{x} + K_{z_2} |r| \tilde{x} \right) + K_{z_3} \left( |\bar{Z}|_F + Z_m \right)^2 \tilde{x} + K_{z_4} \tilde{x}, \\
\]

(4.38)

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

\[
\hat{v} = -T_{x_m} \hat{V}^T \hat{\sigma}' - k T \hat{x} \hat{\tilde{x}}, \\
\hat{w} = -S (\hat{\sigma}' - \hat{\sigma}' V_{x_m}) \hat{x}^2 - k S \hat{x} \hat{\tilde{x}}, \\
\]

(4.39)

(4.40)

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 \(\tilde{x}(t)\) and NN weight estimates \(\hat{V}, \hat{W}\) are UUB, with bounds given by (4.54), (4.55). Moreover, the error \(x(t)\) can be made arbitrarily small by increasing the gain \(K_{\sigma}\).

**Proof.** Select the Lyapunov function candidate

\[
L = L_1 + \frac{1}{2} \hat{x}^T \hat{\tilde{x}} + \frac{1}{2} |r| Tr (\hat{W}^T S^{-1} \hat{W}) + \frac{1}{2} Tr (\hat{V}^T T^{-1} \hat{V}), \\
\]

(4.41)

which weights both errors \(r(t)\) and \(\tilde{x}(t)\), and NN weights estimation errors. Taking derivative

\[
\dot{L} = L_1 + \frac{1}{2} \hat{x}^T \hat{\tilde{x}} + \frac{1}{2} Tr (\hat{W}^T S^{-1} \hat{W}) + \frac{1}{2} Tr (\hat{V}^T T^{-1} \hat{V}), \\
\]

(4.42)

and using (4.6), (4.32) one has

\[
\dot{L} = r^T (f(x) + Y_d + \tau_d + \tau) \\
+ \frac{1}{2} \hat{x}^T \left( -K_s \hat{r} - \hat{V}^T \hat{\sigma}' V_{x_m} - \hat{W}^T \hat{\sigma}' V_{x_m} + w - v_2 \right), \\
+ Tr (\hat{W}^T S^{-1} \hat{W}) + Tr (\hat{V}^T T^{-1} \hat{V}) \\
\]

(4.43)

\[
\dot{L} = r^T (f(x) + Y_d + \tau_d + \tau_{\text{NN}}) - r^T \hat{x} \\
+ \frac{1}{2} \hat{x}^T \left( -K_s \hat{r} - \hat{V}^T \hat{\sigma}' V_{x_m} - \hat{W}^T \hat{\sigma}' V_{x_m} + w - v_2 \right), \\
+ Tr (\hat{W}^T S^{-1} \hat{W}) + Tr (\hat{V}^T T^{-1} \hat{V}) \\
\]

(4.44)

\[
\dot{L} = r^T (f(x) + Y_d + \tau_d + \tau_{\text{NN}}) + Tr (\hat{V}^T T^{-1} \hat{V}) \\
+ \frac{1}{2} \hat{x}^T \left( -K_s \hat{r} + w - v_2 \right), \\
+ k |r| Tr (\hat{W}^T \hat{W} - \hat{W}^T \hat{W}) + k |r| Tr (\hat{V}^T \hat{V}) \\
\]

(4.45)

Applying (4.9) and tuning rules yields

\[
\dot{L} = r^T (\hat{f}(x) + \tau_{\text{NN}}) - r^T \hat{x} + \hat{x}^T (r - K_s \hat{r} + w) \\
+ k |r| Tr (\hat{W}^T \hat{W} - \hat{W}^T \hat{W}) + k |r| Tr (\hat{V}^T \hat{V}) \\
\]

(4.46)

Using the same inequality as for (4.15), expression (4.46) can be bounded as

\[
\dot{L} \leq -K_{\text{NN}} |r|^2 - |r|^2 (f_m + \tau_m) + |r|^2 (f + \tau_d) \\
+ k |r| Tr (\hat{W}^T \hat{W} - \hat{W}^T \hat{W}) - k |r| Tr (\hat{V}^T \hat{V}) \\
- \hat{x}^T K_{z_2} \left( |\bar{Z}|_F + Z_m \right) |r| \hat{x} + k |r| Tr (\hat{V}^T \hat{V}) \\
= -K_{z_2} |r|^2 - |r|^2 (f_m + \tau_m) + |r|^2 (f + \tau_d) \\
\]

(4.47)

Including (4.37) and applying some norm properties, one has

\[
\dot{L} \leq -K_{\text{NN}} |r|^2 - |r|^2 (f_m + \tau_m) + |r|^2 (f + \tau_d) \\
\]

Thus, the $L$ is negative as long as

$$
\tau \leq \frac{k(Z_k + C_1)}{2k} + C_0 \quad (4.54)
$$

$$
\tau > \frac{k(Z_k + C_1)}{2k} + \sqrt{\left(\frac{Z_k + C_1}{2k}\right)^2 + C_0} \quad (4.55)
$$

The first terms of (4.39), (4.40) are modified versions of the standard backpropagation algorithm. The $k$ terms correspond to the $\epsilon$-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 (4.37). The second term is given to ensure the stability due to the error $\tau = \tau_{des} - \tau$ in the backstepping design.

The right-hand side of (4.54) can be taken as a practical bound on the error in the sense that $\tau(t)$ will never stay far above it. Note that the stability radius may be decreased any amount by increasing the gain $K_v$. 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 $K_v$. The NN weight errors are fundamentally bounded in terms of $V_{w_{NN}} W_{NN}$.

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.

V. SIMULATION OF NN BACKLASH COMPENSATOR

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

$$
\dot{x}_1 = x_2
$$

$$
\dot{x}_2 = -\frac{1}{T} x_2 + ma x_1^2 \sin (x_1) + mga \cos (x_1) + \tau \quad (5.1)
$$

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 $S = 30I_{11}$, $T = 50I_6$, $k = 0.01$, where $I_N$ isNxN identity matrix. The robustifying signal gains are $K_{z1} = 2$, $K_{z2} = 2$, $K_{z3} = 2$. The controller parameters are chosen as $L = 5$, $K = 10$, $K_0 = 15$.

The NN has $L = 10$ hidden-layer nodes with sigmoidal activation functions. The first-layer weights $V$ are initialized randomly [20]. 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 $\tau_{\text{des}}$ is implemented as $s + 100$.

Figure 5.1 and Fig. 5.2 show the tracking performance of the closed-loop system when $\phi = K_b \tau + \tau_{\text{des}}$.

This case can be understood as compensation of the backlash just with PD gain. It can be seen that backlash degrades the system performance. Desired trajectory which is shown in the figures is $\sin(t)$.

In the next step we simulated the same nonlinear system with the same backlash but now $\phi = K_b \tau + \tau_{\text{des}} + v_2$. In this case backlash compensator consists of additional robustifying term. Results are shown in the Fig. 5.3 and Fig. 5.4.

In the last step we simulated the complete backlash compensator, as is shown in the Fig. 4.1. The NN is added in the feedforward path, and $\phi = K_b \tau + \tau_{\text{des}} - W^\top \sigma(V^\top x_{\text{in}}) + v_2$. Applying the NN backlash compensator greatly reduces the tracking error. Figure 5.5 and Fig. 5.6 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.

Some of the neural network weights are shown in the Fig. 5.7. One can see that the NN weights $W$ vary through

![Figure 5.1 State $x_1(t)$ and tracking error $e_1(t)$, $\phi = K_b \tau + \tau_{\text{des}}$.](image1)

![Figure 5.2 State $x_2(t)$ and tracking error $e_2(t)$, $\phi = K_b \tau + \tau_{\text{des}}$.](image2)

![Figure 5.3 State $x_1(t)$ and tracking error $e_1(t)$, $\phi = K_b \tau + \tau_{\text{des}} + v_2$.](image3)

![Figure 5.4 State $x_2(t)$ and tracking error $e_2(t)$, $\phi = K_b \tau + \tau_{\text{des}} + v_2$.](image4)
time, tuning 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.

**VI. 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 an efficient way of improving the tracking performance of nonlinear systems with backlash.

**REFERENCES**


F.L. Lewis was born in Würzburg, Germany, subsequently studying in Chile and Gordonstoun School in Scotland. He obtained the Bachelor’s Degree in Physics/Electrical Engineering and the Master’s of Electrical Engineering Degree at Rice University in 1971. He spent six years in the U.S. Navy, serving as Navigator aboard the frigate USS Trippe (FF-1075), and Executive Officer and Acting Commanding Officer aboard USS Salinan (ATF-161). In 1977 he received the Master’s of Science in Aeronautical Engineering from the University of West Florida. In 1981 he obtained the Ph.D. degree at The Georgia Institute of Technology in Atlanta, where he was employed as a professor from 1981 to 1990 and is currently an Adjunct Professor. He is a Professor of Electrical Engineering at The University of Texas at Arlington, where he was awarded the Moncrief-O’Donnell Endowed Chair in 1990 at the Automation and Robotics Research Institute.

Dr. Lewis has studied the geometric, analytic, and structural properties of dynamical systems and feedback control automation. His current interests include robotics, intelligent control, neural and fuzzy systems, nonlinear systems, and manufacturing process control. He is the author/co-author of 2 U.S. patents, 124 journal papers, 20 chapters and encyclopedia articles, 210 refereed conference papers, seven books: Optimal Control, Optimal Estimation, Applied Optimal Control and Estimation, Aircraft Control and Simulation, Control of Robot Manipulators, Neural Network Control, High-Level Feedback Control with Neural Networks and the IEEE reprint volume Robot Control. Dr. Lewis is a registered Professional Engineer in the State of Texas and was selected to the Editorial Boards of *International Journal of Control, Neural Computing and Applications, and Int. J. Intelligent Control Systems*. He is currently an Editor for the flagship journal Automatica. He is the recipient of an NSF Research Initiation Grant and has been continuously funded by NSF since 1982. Since 1991 he has received $1.8 million in funding from NSF and upwards of $1 million in SBIR/industry/state funding. He has received a Fulbright Research Award, the American Society of Engineering Education F.E. Terman Award, three Sigma Xi Research Awards, the UTA Halliburton Engineering Research Award, the UTA University-Wide Distinguished Research Award, the ARRI Patent Award, various Best Paper Awards, the IEEE Control Systems Society Best Chapter Award (as Founding Chairman), and the National Sigma Xi Award for Outstanding Chapter (as President). He was selected as Engineer of the year in 1994 by the Ft. Worth IEEE Section and is a Fellow of the IEEE. He was appointed to the NAE Committee on Space Station in 1995 and to the IEEE Control Systems Society Board of Governors in 1996. In 1998 he was selected as an IEEE Control Systems Society Distinguished Lecturer. He is a Founding Member of the Board of Governors of the Mediterranean Control Association.

Rastko R. Selmic was born in Belgrade, Serbia, Yugoslavia in 1970. In 1994 he received his Bachelor of Science degree in Electrical Engineering from University of Belgrade. In 1997, he received his Master of Science degree in Electrical Engineering from The University of Texas at Arlington. Since 1995, he has been a graduate research assistant at Automation and Robotics Research Institute. In 2000, he received his Doctor of Philosophy degree in Electrical Engineering from The University of Texas at Arlington, under supervision of Dr. Frank Lewis. His research interest is in nonlinear control, adaptive control, neural networks and fuzzy logic, compensation of actuator nonlinearities using intelligent control techniques.

He has a U.S. patent pending “Method For Backlash Compensation Using Neural Networks.” He is the recipient of the first prize at the IEEE Fort Worth Section Graduate Paper Contest for 1999, finalist for the Best Paper Award at 1998 IEEE International Conference on Control Applications. He is recognized as University Scholar at Academic Excellence Convocation in 1998, received ARRI Best Student Paper Award for 1997, and scholarship from Signologic in 1996 for the results achieved. He is a member of the IEEE and Sigma Xi.