Robust Adaptive Nonlinear Control using Single Hidden Layer Neural Networks

Flavio Nardi * and Anthony J. Calise†
School of Aerospace Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0150

Abstract

In this paper we develop an adaptive dynamic inverting controller with guaranteed closed loop stability for partially or completely unknown nonlinear non affine dynamic systems. We assume full state feedback and no zero dynamics. A single hidden layer neural network is used to approximate the inverse map, and a stable adaptive scheme is used to update on-line the neural network weights. Stability is guaranteed by introducing a robust adaptive bound. The performance of the adaptive scheme is demonstrated in a tracking task controller design and simulation for the nonlinear Van der Pol oscillator.

Introduction

In this last decade there has been a great deal of research in the control of highly uncertain dynamic systems. The leading methodology has been adaptive nonlinear control. Its application to linearly parameterized system uncertainty has developed into systematic approaches based on either feedback linearization or back-stepping [11, 15, 25]. Extensions to nonlinearly parameterized uncertainties have been possible by employing artificial neural networks in the feedback loop [8]. Adaptive control using on-line function approximators for feedback linearizable systems has proven to be a very effective way to design controllers based on a very crude knowledge of the system dynamics.

Adaptive control of feedback linearizable systems was first introduced by Sastry et al. [24] and Taylor et al. [27]. Neural networks (NNs) for identification and control were first proposed by Narendra and Parthasarathy [17]. Results in this initial research were limited to simulation, and no proofs of stability in the closed loop were provided. First results in proving stability of on-line feedback linearizing adaptive neural network augmented controllers were obtained by Chen and Khalil [4] for discrete time systems, and by Chen and Liu [5] for continuous time systems. Sanner and Slotine [22] proposed the use of radial basis function neural networks in conjunction with the computed torque method approach. A stability proof is provided using sliding mode control. Lewis et al. [28, 13, 14] developed novel adaptive control update laws for an on-line Single Hidden Layer (SHL) neural network based on Lyapunov analysis. These results were limited to affine in control nonlinear systems. Calise et al. [10, 3] removed the affine in control restriction by developing a dynamic inversion based control architecture with linearly parameterized neural networks in the feedback path to compensate for the inversion error introduced by an approximate inverse. Extensions of this work to SHL neural networks can be found in [16]. Flight testing has been carried out using this control architecture on the X 36 aircraft, demonstrating both greater performance than the baseline dynamic inverting controller, and superior fault tolerance ability [2].

Relevant contributions in adaptive neural network theory were also introduced by Rovithakis and Christodoulou [21] and by Polycarpou [18, 19]. In particular, Polycarpou introduced the use of a robust adaptive gain to compensate for neglected higher order terms in the Taylor series expansion of the neural network output. This expansion is necessary for implementable update laws, and the advantage of the adaptive bounding approach proposed by Polycarpou is that only the functional dependence of these higher order terms is needed, not their magnitude. Specifically, there is no need to know the upper bound on the neural network weight matrices.

More recently, Zhang et al. [29] have extended the use of Lewis' SHL adaptive laws to nonaffine systems and analyzed transient performance. A very interesting approach using piecewise linear approximators has been presented by Choi and Farrel [6]. Also, Rovithakis [20] introduced a dynamic neural network for inverse control of affine systems by developing adaptive laws based on Lyapunov analysis.

In this paper we extend the results in [19] to non affine nonlinear dynamical systems and specialize the function approximator to the SHL neural network. We explicitly construct adaptive update laws from Lya-

*Graduate Research Assistant, Student Member IEEE.
†Professor, Member IEEE.

punov analysis. Boundedness of the signals is proven by introducing a robust adaptive control component.

The paper is organized as follows: first we present the dynamic system and dynamic inversion for non affine systems; next we introduce the single hidden layer neural network model and the stability analysis. Then we present simulation results of the proposed controller in a tracking task. Summary and concluding remarks are given at the end.

Dynamic inversion and derivation of the error dynamics

Consider the following nonlinear system in Brunovsky form:

$$\dot{x}_n \quad = \quad f(\mathbf{x}, u) + d$$

where $\mathbf{x} \in R^n$, $u \in R$, f is sufficiently smooth, and $d \in L^\infty$, i.e. $|d| \leq \phi^*$, $\phi^* > 0$.

Assumption 1. There exist positive constants f^L, f^U, H such that

$$0 < f^L \leq \frac{\partial f(\mathbf{x}, u)}{\partial u} \leq f^U, \tag{1}$$

and

$$\left| \frac{d}{dt} \frac{\partial f(\mathbf{x}, u)}{\partial u} \right| \leq H < \frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)} f^L, \tag{2}$$

for $(\mathbf{x}, u) \in \Omega \times R$, with $\Omega \subset R^n$, where Q, P are positive definite matrices that will be defined in the sequel. The condition in Eq.(2) is needed in the stability analysis.

Consider the following state dependent transformation

$$\begin{aligned} \dot{x}_n &= v \\ v &= f(\mathbf{x}, u), \end{aligned} \tag{3}$$

where v is commonly referred to as the pseudo control.

The pseudo control is chosen in this derivation as a linear operator. In general, it may also be nonlinear, for example including a sliding mode component [23, 26].

The transformation in Eq.(3) is defined locally by invoking the implicit function theorem [1]. Since the pseudo control v is in general not a function of the control u but rather a state dependent operator, by Assumption 1 we have

$$\frac{\partial [v - f(\mathbf{x}, u)]}{\partial u} \neq 0. \tag{4}$$

The fact that the expression in Eq.(4) is non singular implies that in a neighborhood of every $(\mathbf{x}, u) \in \Omega \times R$ there exists an implicit function $\alpha(\mathbf{x}, v)$ such that

$$v - f(\mathbf{x}, \alpha(\mathbf{x}, v)) = 0. \tag{5}$$

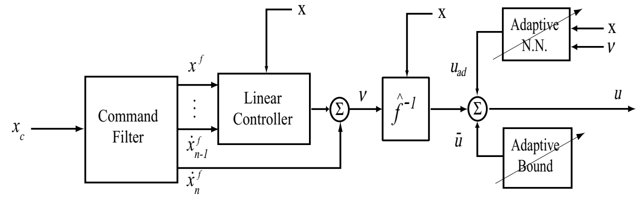

Fig. 1 Controller Architecture.

Since Assumption 1 holds in a domain $\Omega \times R$, we could consider the union of all such neighborhoods and extend the existence of the transformation to the entire domain. Let

$$u^* \triangleq \alpha(\mathbf{x}, v) = f^{-1}(\mathbf{x}, v). \tag{6}$$

Before proceeding to the construction of the error equation consider the following remark.

Remark 1. For systems that are affine in the control as

$$\dot{x}_n \quad = \quad f(\mathbf{x}) + g(\mathbf{x})u$$

the ideal feedback linearizing law is given by

$$u^* = \frac{-f(\mathbf{x}) + v}{g(\mathbf{x})}.$$

This feedback linearizing law is defined for all $(\mathbf{x}, u) \in \Omega \times R$ as long as g is bounded away from zero, whereas in the non affine case we can only invoke the implicit function theorem for local existence (although we could consider the union of all neighborhoods to extend the result). The analysis in this paper is therefore equally applicable to affine in control systems.

With reference to Fig. 1 define the tracking error $e = x^f - x \triangleq e_1$; the error dynamics can be written in the following form

$$\begin{aligned} \dot{e}_i &= e_{i+1}, \qquad i = 1, \ldots, n-1 \\ \dot{e}_n &= \dot{x}_n^f - f(\mathbf{x}, u) - d. \end{aligned}$$

By invoking the mean value theorem there exists a real number λ,with $0 < \lambda < 1$, such that

$$f(\mathbf{x}, u) = f(x, u^*) + \left. \frac{\partial f(\mathbf{x}, u)}{\partial u} \right|_{u = u_\lambda} (u - u^*),$$

with $u_\lambda = \lambda u + (1 - \lambda) u^*$. Introduce the short hand notation for the partial derivative $f_u \triangleq \left. \frac{\partial f}{\partial u} \right|_{u = u_\lambda}$. The n^{th} equation in the error dynamics becomes

$$\dot{e}_n \quad = \quad \dot{x}_n^f - f(\mathbf{x}, u^*) - f_u (u - u^*) - d.$$

Based on the transformation introduced in Eq.(3), we know that for $u^* = f^{-1}(\mathbf{x}, v)$ we have $f(\mathbf{x}, u^*) = v$. Using this fact in the error dynamics

$$\dot{e}_n \;=\; \dot{x}_n^f - v - f_u(u - u^*) - d.$$

Let the control u be defined as follows:

$$u \triangleq \hat{f}^{-1}(\mathbf{x}, v) + u_{ad} + \bar{u}, \qquad (7)$$

where $\hat{f}^{-1}(\mathbf{x}, v)$ is an arbitrarily poor approximation of the unknown function $f^{-1}(\mathbf{x}, v)$, u_{ad} is a neural network based adaptive signal, and $\bar{u}$ is a robust adaptive term.

Assumption 2. When introducing an approximate inversion model, experience has shown that it is best to satisfy the following condition

$$\mathrm{sign}\left(\frac{\partial \hat{f}(\mathbf{x}, u)}{\partial u}\right) \equiv \mathrm{sign}\left(\frac{\partial f(\mathbf{x}, u)}{\partial u}\right)$$

for all $(\mathbf{x}, u) \in \Omega \times R$.

Remark 2. The choice of an approximate inverse in Eq.(7) is not required. In the absence of any information about the system dynamics except for the sensitivity as of Assumption 2, the control may as well be defined as

$$u \triangleq v + u_{ad} + \bar{u}. \qquad (8)$$

The controller architecture with the definition of the control u using an approximate inverse as in Eq.(7) is shown in Fig. 1. In this block diagram we have included the command filter, a linear transfer function that generates the filtered command and its derivatives that are needed in the construction of the pseudo control v. Substituting the control law and the expression for u^* in the n^{th} equation of the error dynamics:

$$\dot{e}_n \;=\; \dot{x}_n^f - v - f_u\left[u_{ad} - \Delta(\mathbf{x}, v) + \bar{u}\right] - d,$$

where the inversion error $\Delta(\mathbf{x}, v)$ is defined as

$$\Delta(\mathbf{x}, v) \triangleq f^{-1}(\mathbf{x}, v) - \hat{f}^{-1}(\mathbf{x}, v).$$

The pseudo control v is designed as

$$v \triangleq \dot{x}_n^f + k_{n-1}e_{n-1} + \cdots + k_1 e_1 + \bar{v}, \qquad (9)$$

where $\bar{v}$ is a nonlinear term introduced to reject the disturbance d, such that the error dynamics can be written in matrix form

$$\dot{\mathbf{e}} \;=\; A\mathbf{e} - \mathbf{b}\left(f_u\left[u_{ad} - \Delta(\mathbf{x}, v) + \bar{u}\right] + \bar{v} + d\right), \qquad (10)$$

for $\mathbf{e} = [\; e_1 \;\; e_2 \;\; \cdots \;\; e_n \;]^T$, $\mathbf{b} = [\; 0 \;\; 0 \;\; \cdots \;\; 1 \;]^T$, and A a Hurwitz matrix with the following structure:

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -k_1 & -k_2 & -k_3 & \cdots & -k_{n-1} \end{bmatrix}.$$

Since A is Hurwitz, a unique positive definite solution P to the following Lyapunov equation exists

$$A^T P + PA = -Q, \qquad (11)$$

where Q is any positive definite matrix.

Neural Network Augmentation and Stability Analysis

In this section we analyze the neural network approximation of the inversion error function $\Delta(\mathbf{x}, v)$ and we present the uniform boundedness analysis of all signals in the closed loop.

The input-output map of a SHL perceptron can be written in matrix form as [12]

$$y = M^T \sigma(N^T \bar{x}),$$

where the weight matrices include the bias in the first row. The squashing function σ is defined as:

$$\sigma(z) = \frac{1 - e^{-az}}{1 + e^{-az}}.$$

The factor a is known as the activation potential.

Based on the universal approximation property of SHL NN [9, 7] there exists a set of NN weights M and N such that the nonlinear approximation error $\Delta(\mathbf{x}, v)$ can be approximated by its output as

$$M^T \sigma(N^T \bar{x}) = \Delta(\mathbf{x}, v) - \epsilon(\bar{x})$$

for all $(\mathbf{x}, v) \in D,\;\; D \subset R^{n+1}$, and the approximation error is bounded as

$$|\epsilon(\bar{x})| \leq \psi_\epsilon^*$$

where $\psi_\epsilon^* > 0$ is an unknown constant, and $\bar{x} = [\; \mathbf{x} \;\; v \;]^T$. In the above expressions M and N indicate constant parameter values that minimize $|\epsilon(\bar{x})|$. These values are not necessarily unique.

Let $\hat{M}$ and $\hat{N}$ be the estimates of M and N respectively, and M_0, N_0 the initial value of these estimates. Let u_{ad} be the output of a SHL neural network

$$u_{ad} \triangleq \hat{M}^T \sigma(\hat{N}^T \bar{x}). \qquad (12)$$

Then the error dynamics in Eq.(10) can be written as

$$\begin{aligned}\dot{\mathbf{e}} &= A\mathbf{e} - \mathbf{b}\Big(f_u\Big[\hat{M}^T\sigma(\hat{N}^T\bar{x}) \\ &\quad - M^T\sigma(N^T\bar{x}) + \bar{u} - \epsilon(\bar{x})\Big] + \bar{v} + d\Big),\end{aligned}$$

Define $\tilde{M} = \hat{M} - M$, $\tilde{N} = \hat{N} - N$. In order to back-propagate the estimation error through the NN hidden layer consider the Taylor series expansion of $\sigma(N^T\bar{x})$ about $\hat{N}^T\bar{x}$:

$$\sigma(N^T\bar{x}) = \sigma(\hat{N}^T\bar{x}) - \sigma'(\hat{N}^T\bar{x})\tilde{N}^T\bar{x} + \mathcal{O}(-\tilde{N}^T\bar{x})^2.$$

Following the procedure presented in [13, 29] it can be shown that the following holds:

$$\begin{aligned}&\hat{M}^T\sigma(\hat{N}^T\bar{x}) - M^T\sigma(N^T\bar{x}) = \\ &\qquad \tilde{M}^T(\hat{\sigma} - \hat{\sigma}'\hat{N}^T\bar{x}) + \hat{M}^T\hat{\sigma}'\tilde{N}^T\bar{x} + w\end{aligned}$$

where $\hat{\sigma} \triangleq \sigma(\hat{N}^T\bar{x})$, $\hat{\sigma}'$ is the Jacobian of $\hat{\sigma}$, and the higher order terms included in w are bounded as follows

$$|w| \le \|N\|_F\|\bar{x}\hat{M}^T\hat{\sigma}'\|_F + \|M\|\|\hat{\sigma}'\hat{N}^T\bar{x}\| + \|M\|_1,$$

where "F" stands for Frobenius norm and "1" for the 1 norm. This bound can be re-written as follows:

$$|w| \le \psi_w^* s_w(\hat{N}, \hat{M}, \bar{x}),$$

where

$$\begin{aligned}\psi_w^* &\triangleq \max\{\|N\|_F, \|M\|, \|M\|_1\}, \\ s_w(\hat{N}, \hat{M}, \bar{x}) &\triangleq \|\bar{x}\hat{M}^T\hat{\sigma}'\|_F + \|\hat{\sigma}'\hat{N}^T\bar{x}\| + 1.\end{aligned}$$

Note that ψ_w is an unknown coefficient and s_w is a known function. Substituting the Taylor series expansion in the expression for the error dynamics we obtain

$$\begin{aligned}\dot{\mathbf{e}} &= A\mathbf{e} - \mathbf{b}\Big(f_u\Big[\tilde{M}^T(\hat{\sigma} - \hat{\sigma}'\hat{N}^T\bar{x}) \\ &\quad + \hat{M}^T\hat{\sigma}'\tilde{N}^T\bar{x} + \bar{u} + w - \epsilon(\bar{x})\Big] + \bar{v} + d\Big), \quad (13)\end{aligned}$$

Define

$$\zeta \triangleq \mathbf{e}^T P\mathbf{b}. \quad (14)$$

In the proof of stability we will use the following result:

Lemma 1. Let M,$\hat{M}$,M_0, and $\tilde{M}$ be matrices in $R^{n\times m}$ with $\tilde{M} \triangleq \hat{M} - M$. Then the following fact holds:

$$\begin{aligned}\mathrm{tr}\left[\tilde{M}^T(\hat{M} - M_0)\right] &= \frac{1}{2}\|\tilde{M}\|_F^2 + \frac{1}{2}\|\hat{M} - M_0\|_F^2 \\ &\quad - \frac{1}{2}\|M - M_0\|_F^2\end{aligned}$$

Proof: Direct expansion and manipulation of the right hand side is sufficient to prove the equality. ■

Theorem 1. Consider the domain Ω containing the origin. Then the feedback control law given by

$$u = \hat{f}^{-1}(\mathbf{x}, v) + u_{ad} + \bar{u},$$

where $u_{ad} = \hat{M}^T\hat{\sigma}(\hat{N}^T\bar{x})$, $\bar{u} = \psi s^* \tanh\left(\frac{2\zeta s^*}{\delta}\right)$, $\bar{v} = \phi\,\mathrm{sgn}(\zeta)$, v and $\hat{f}^{-1}$ are defined in the previous section, with the following adaptive laws

$$\begin{aligned}\dot{\hat{N}} &= G\left[2\bar{x}\zeta\hat{M}^T\hat{\sigma}' - \lambda_N|\zeta|(\hat{N} - N_0)\right] \\ \dot{\hat{M}} &= F\left[2(\hat{\sigma} - \hat{\sigma}'\hat{N}^T\bar{x})\zeta - \lambda_M|\zeta|(\hat{M} - M_0)\right] \\ \dot{\psi} &= \gamma_\psi\left[2\zeta s^*\tanh\left(\frac{2\zeta s^*}{\delta}\right) - \lambda|\zeta|(\psi - \psi_0)\right] \\ \dot{\phi} &= \gamma_\phi\left[2|\zeta| - \lambda_\phi|\zeta|(\phi - \phi_0)\right],\end{aligned}$$

for $s^* \triangleq 1 + s_w$, $\psi^*_{\max} \triangleq \max\{\psi_\epsilon^*, \psi_w^*\}$, $F, G, \gamma_\psi, \gamma_\phi > 0$, $\lambda_M, \lambda_N, \lambda, \delta > 0$, guarantees that all signals in the closed loop system are uniformly bounded and that the tracking error $\mathbf{e}$ is uniformly ultimately bounded.

Proof: Let $\tilde{\psi} \triangleq \psi - \psi^*_{\max}$ and $\tilde{\phi} \triangleq \phi - \phi^*$. Let ψ_0, ϕ_0 denote the initial estimates of ψ and ϕ respectively. Consider the following Lyapunov function candidate V

$$\begin{aligned}V &= \frac{\mathbf{e}^T P\mathbf{e}}{f_u} + \frac{1}{2}\mathrm{tr}(\tilde{M}^T F^{-1}\tilde{M}) + \frac{1}{2}\mathrm{tr}(\tilde{N}^T G^{-1}\tilde{N}) \\ &\quad + \frac{1}{2}\tilde{\psi}\gamma_\psi^{-1}\tilde{\psi} + \frac{1}{2f^U}\tilde{\phi}\gamma_\phi^{-1}\tilde{\phi}. \quad (15)\end{aligned}$$

The derivative of V along the trajectories of Eq.(13) after substituting the adaptive laws is:

$$\begin{aligned}\dot{V} &= -\frac{\mathbf{e}^T Q\mathbf{e}}{f_u} - \frac{\mathbf{e}^T P\dot{f}_u\mathbf{e}}{f_u^2} - \lambda_M|\zeta|\mathrm{tr}\tilde{M}^T(\hat{M} - M_0) \\ &\quad - \lambda_N|\zeta|\mathrm{tr}\tilde{N}^T(\hat{N} - N_0) - 2\zeta\left[\bar{u} + w - \epsilon(\bar{x})\right] \\ &\quad + \tilde{\psi}\left[2\zeta s^*\tanh\left(\frac{2\zeta s^*}{\delta}\right) - \lambda|\zeta|(\psi - \psi_0)\right] \\ &\quad - \frac{2\zeta}{f_u}[\bar{v} + d] + \frac{\tilde{\phi}}{f^U}\left[2|\zeta| - \lambda_\phi|\zeta|(\phi - \phi_0)\right].\end{aligned}$$

The derivative is bounded by

$$\begin{aligned}\dot{V} &\le -\frac{\mathbf{e}^T[\lambda_{\min}(Q) - H\lambda_{\max}(P)/f^L]\mathbf{e}}{f_u} \\ &\quad - \lambda_M|\zeta|\mathrm{tr}\tilde{M}^T(\hat{M} - M_0) - \lambda_N|\zeta|\mathrm{tr}\tilde{N}^T(\hat{N} - N_0) \\ &\quad - \lambda|\zeta|\tilde{\psi}(\psi - \psi_0) - \lambda_\phi|\zeta|\tilde{\phi}(\phi - \phi_0) \\ &\quad + 2|\zeta|\phi^*\left[\frac{1}{f^L} - \frac{1}{f^U}\right] + \psi^*_{\max}\kappa\delta,\end{aligned}$$

where we have used the fact that $0 \leq |\eta| - \eta \tanh\left(\frac{\eta}{\delta}\right) \leq \kappa\delta$, and $\kappa = 0.2785$ (refer to [18, 19] for more details). By using Lemma 1, it is straightforward to show that the following bound holds:

$$
\begin{aligned}
\dot{V} \quad \leq \quad & -[\lambda_{\min}(Q) - H\lambda_{\max}(P)/f^L]\frac{\mathbf{e}^T\mathbf{e}}{f_u} \\
& + \frac{\lambda_M}{2}|\zeta|\|M - M_0\|_F^2 + |\zeta|\frac{\lambda_N}{2}\|N - N_0\|_F^2 \\
& + \frac{\lambda}{2}|\zeta||\psi^*_{\max} - \psi_0|^2 + \frac{\lambda_\phi}{2}|\zeta||\phi^* - \phi_0|^2 \\
& + 2|\zeta|\phi^*\left[\frac{1}{f^L} - \frac{1}{f^U}\right] + \psi^*_{\max}\kappa\delta.
\end{aligned}
$$

Define

$$
\begin{aligned}
\bar{Z} \quad = \quad & \frac{\lambda_M}{2}\|M - M_0\|_F^2 + \frac{\lambda_N}{2}\|N - N_0\|_F^2 \\
& + \frac{\lambda}{2}|\psi^*_{\max} - \psi_0|^2 + \frac{\lambda_\phi}{2}|\phi^* - \phi_0|^2 \\
& + 2\phi^*\left[\frac{1}{f^L} - \frac{1}{f^U}\right],
\end{aligned}
$$

then

$$
\begin{aligned}
\dot{V} \quad \leq \quad & -[\lambda_{\min}(Q) - H\lambda_{\max}(P)/f^L]\frac{\|\mathbf{e}^2\|}{f_u} \\
& + \|\mathbf{e}\|\|P\mathbf{b}\|\bar{Z} + \psi^*_{\max}\kappa\delta.
\end{aligned}
$$

Define $\Xi \triangleq \frac{[\lambda_{\min}(Q) - H\lambda_{\max}(P)/f^L]}{f_u}$. This last bound can be reduced to

$$\dot{V} < 0$$

provided

$$\|\mathbf{e}\| \quad > \quad \frac{\|P\mathbf{b}\|\bar{Z} + \sqrt{(\|P\mathbf{b}\|\bar{Z})^2 + 4\Xi\psi^*_{\max}\kappa\delta}}{2\Xi}. \tag{16}$$

This condition implies uniform ultimate boundedness of the tracking error and uniform boundedness of $\tilde{M}, \tilde{N}, \tilde{\psi}, \tilde{\phi}$. The compact set in Eq.(16) can be made as small as desired by increasing the gains in the linear control design. A proper choice will ensure this set to be contained in the domain D. ■

Simulation Results

To illustrate the performance of the proposed adaptive controller, we consider a Van der Pol oscillator with the following dynamics

$$
\begin{aligned}
\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} \quad = \quad & \begin{bmatrix} 0 & 1 \\ -\kappa & \alpha \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\
& + \begin{bmatrix} 0 \\ 1 \end{bmatrix}[(x_1^2 + x_2^2)\sigma(u) - \beta x_1^2 x_2] + d,
\end{aligned}
$$

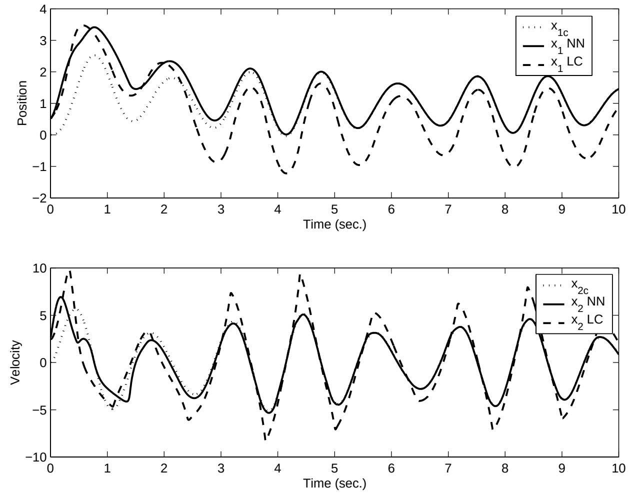

Fig. 2 Tracking Performance with/without NN.

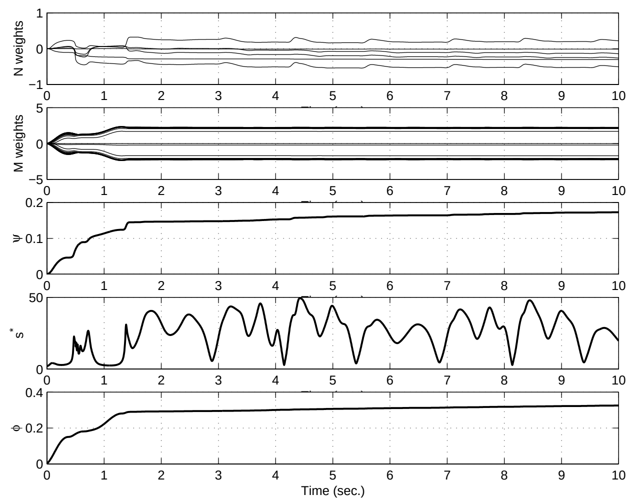

Fig. 3 History of Adaptive Parameters and s^*.

where $\sigma(u) = (1 - e^{-u})/(1 + e^{-u})$ and $d = 0.2$ is the disturbance. No approximate inversion is assumed, which amounts to letting $\hat{f}^{-1}(\mathbf{x}, v) = v$, reducing the control law to

$$u = \dot{x}_{2c} + k_d\tilde{x}_2 + k_p\tilde{x}_1 + u_{ad} + \bar{u}, \tag{17}$$

where $\tilde{x}_i \triangleq x_{ic} - x_i$, the subscript c denoting filtered commands. The parameters in the dynamics are $\alpha = 2\zeta\omega$, $\kappa = \omega^2$, and $\beta = |\alpha|$, for $\omega = 0.4\pi$. The gains are chosen as follows: $k_p = 1$; $k_d = 2$; $F, G = I$; $\gamma_\psi = 0.01$; $\lambda_M, \lambda_N = 0.01$; $\lambda_\psi = 0.1$; $\lambda_\phi = 1$; $\gamma_\phi = 0.1$.

We employed 27 hidden layer neurons; the vector of inputs to the neural network $\bar{x}$ is chosen to include the states and filtered commands. The neural network weights are initialized at zero, and the hyperbolic tangent functions of each neuron are shifted by a constant amount, i.e.

$$\sigma(\hat{N}^T\bar{x} - \xi), \tag{18}$$

where ξ is a vector of constant values that offsets each neuron. This values of ξ were chosen between -2.5 and 2.5 with 0.2 spacing. The motivation behind shifting each neuron by a different constant value comes from the fact that distributing them over a certain domain will cause each neuron to exhibit a different activation. Although the inner layer weights' bias can shift the neurons, offsetting them causes the weights to move in different regions of the weight space. The transient learning involves those neurons that are closer to the activation region, and hence, although the gradient of all neurons have the same sign, its magnitude varies greatly due to the offset.

Fig. 2 compares the performance of the baseline controller (no NN) and the NN augmented controller. The disturbance was not included in the linear control case. Fig. 3 illustrates the weight histories, which reach a nearly steady state value after 1 second, the history of the adaptive bound gain ψ, the magnitude function s^* used in bounding the higher order terms in the Taylor series expansion, and the history of the adaptive bound on the disturbance d. It is worth noting that this estimate is slightly bigger than the true value (0.3 vs. 0.2), thus ensuring disturbance rejection.

Conclusions

In this paper we have presented an adaptive feedback linearizing controller architecture that is applicable to the control of nonlinear non affine dynamical systems. A single hidden layer neural network is used to estimate on-line the inverting control law. Update laws are derived from Lyapunov analysis. The performance of the controller is demonstrated in tracking control of the Van der Pol oscillator.

References

[1] R.G. Bartle, The Elements of Real Analysis, Wiley & Sons, 1964.

[2] A.J. Calise, S. Lee, and M. Sharma, Direct Adaptive reconfigurable Control of a Tailless Fighter Aircraft, Proceedings of the AIAA Guidance,Navigation and Control Conference, also submitted to the Journal of Guidance, Control, and Dynamics (1998).

[3] A.J. Calise and R.T. Rysdyk, Nonlinear Adaptive Flight Control using Neural Networks, IEEE Control System Magazine **18** (1998), no. 6, 14–25.

[4] F.C. Chen and H.K. Khalil, Adaptive Control of Nonlinear Systems using Neural Networks, International Journal of Control **55** (1992), no. 6, 1299–1317.

[5] F.C. Chen and C.C. Liu, Adaptively controlling nonlinear continuous-time systems using multilayer neural networks, IEEE Trans. Autom. Contr. **39** (1994), no. 6, 1306–1310.

[6] J.Y. Choi and J. Farrell, Nonlinear adaptive control ausing networks of piecewise linear approximators, Proceedings of IEEE Conf. Dec. and Contr. (1999), 1671–1676.

[7] K. Funahashi, On the Approximate Realization of Continuous Mappings by Neural Networks, Neural Networks **2** (1989), 183–192.

[8] M.M. Gupta and D.H. Rao, Neuro-control systems: theory and applications, IEEE Press, 1994.

[9] K. Hornik, M. Stinchcombe, and H. White, Multilayer Feedforward Networks are Universal Approximators, Neural Networks **2** (1989), 359–366.

[10] B.S Kim and A.J. Calise, Nonlinear Flight Control Using Neural Networks, Journal of Guidance,Control and Dynamics **20** (1997), no. 1, 26–33.

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

[12] F.L. Lewis, S. Jagannathan, and A. Yesildirek, Neural Network Control of Robot Manipulators and Nonlinear Systems, Taylor & Francis, 1998.

[13] F.L. Lewis, A. Yesildirek, and K. Liu, Multilayer Neural-Net Robot Controller with Guaranteed Tracking Performance, IEEE Transactions on Neural Networks **7** (1996), no. 2, 1–12.

[14] F.L. Lewis, A. Yesildirek, and K. Liu, Neural net robot controller: structure and stability proofs, Journal of Intelligent and Robotic System **12** (1996), 277–299.

[15] R. Marino and P. Tomei, Nonlinear Control Design: Geometric, Adaptive, & Robust, Prentice Hall, New Jersey, 1995.

[16] M.B. McFarland and A.J. Calise, Multilayer neural networks and adaptive control of agile anti-air missile, Proceedings of the AIAA Guidance,Navigation and Control Conference **AIAA 97-3540** (1997).

[17] K.S. Narendra and K. Parthasarathy, Identification and control of dynamical systems using neural networks, IEEE Transactions on Neural Networks **1** (1990), 4–27.

[18] M.M. Polycarpou, Stable adaptive neural control scheme for nonlinear systems, IEEE Trans. Autom. Contr. **41** (1996), no. 3, 447–451.

[19] M.M. Polycarpou and M.J. Mears, Stable adaptive tracking of uncertain systems using nonlinearly parametrized on-line approximators, International Journal of Control **70** (1998), no. 3, 363–384.

[20] G.A. Rovithakis, Nonlinear Adaptive Control in the Presence of Unmodelled Dynamics using Neural Networks, Proceedings of IEEE Conf. Dec. and Contr. (1999), 2150–2155.

[21] G.A. Rovithakis and M.A. Chritodoulou, Adaptive Control of Unknown Plants Using Dynamical Neural Networks, IEEE Trans. Sys. Man and Cybern. **24** (1994), no. 3, 400–411.

[22] R. Sanner and J.J. Slotine, Gaussian networks for direct adaptive control, IEEE Transactions on Neural Networks **3** (1992), no. 6, 837–864.

[23] S.S. Sastry, Nonliner Systems, Springer, 1999.

[24] S.S. Sastry and A. Isidori, Adaptive Control of Linearizable Systems, IEEE Trans. Autom. Contr. **34** (1989), no. 11, 1123–1131.

[25] R. Sepulchre, M. Jankovic, and P. Kokotovic, Constructive Nonlinear Control, Springer-Verlag, 1997.

[26] J.T. Spooner and K.M. Passino, Decentralized adaptive control of nonlinear systems using radial basis neural networks, IEEE Trans. Autom. Contr. **44** (1999), no. 11, 2050–2057.

[27] D.G. Taylor, P.V. Kokotovic, R. Marino, and I. Kanellakopoulos, Adaptive Regulation of Nonlinear Systems with Unmodeled Dynamics, IEEE Trans. Autom. Contr. **34** (1989), no. 4, 405–412.

[28] A. Yesildirek and F.L. Lewis, Feedback Linearization using Neural Networks, Automatica **31** (1995), no. 11, 1659–1664.

[29] T. Zhang, S.S. Ge, and C.C. Hang, Design and performance analysis of a direct adaptive controller for nonlinear systems, Automatica **35** (1999), 1809–1817.