A Stable Neural Network-Based Observer With Application to Flexible-Joint Manipulators

Farzaneh Abdollahi, *Student Member, IEEE*, H. A. Talebi, *Member, IEEE*, and Rajnikant V. Patel, *Fellow, IEEE*

***Abstract*—A stable neural network (NN)-based observer for general multivariable nonlinear systems is presented in this paper. Unlike most previous neural network observers, the proposed observer uses a *nonlinear-in-parameters* neural network (NLPNN). Therefore, it can be applied to systems with higher degrees of nonlinearity without any *a priori* knowledge about system dynamics. The learning rule for the neural network is a novel approach based on the modified backpropagation (BP) algorithm. An e-modification term is added to guarantee robustness of the observer. No strictly positive real (SPR) or any other strong assumption is imposed on the proposed approach. The stability of the recurrent neural network observer is shown by Lyapunov's direct method. Simulation results for a flexible-joint manipulator are presented to demonstrate the enhanced performance achieved by utilizing the proposed neural network observer.**

***Index Terms*—Flexible joint manipulators, neural networks (NN), nonlinear observer.**

I. Introduction

THE state of a process specifies its behavior and many control schemes such as inverse dynamics and feedback linearization rely on the availability of the states. However, in many practical systems only the input and output of a system are measurable. Therefore, estimating the states of the system plays a crucial role in monitoring the process, detecting and diagnosing of faults, and achieving better performance. Furthermore, most practical systems are nonlinear, and using linearization or quasi-linearization methods limits the estimation accuracy to a small dynamic range. Several conventional nonlinear observers have been suggested during the past decade. Some of these, such as high-gain observers and sliding mode observers [1]–[3], are only applicable to systems with specific model structures. Furthermore, most of them rely on completely knowing the system nonlinearities *a priori*. Several nonlinear observers have also been suggested for specific systems. However, for most practical processes, defining an exact model is a hard task or is not possible at all. Robot manipulators with flexible joints or links are good examples of such systems. Flexibility in a joint causes extreme difficulty in modeling manipulator dynamics and becomes a potential source of uncertainty that can degrade the performance of the manipulator and in some cases can even destabilize the system [4]. Thus, model-based observers cannot be used for such systems.

The capability of neural networks (NN) for identification, observation and control of nonlinear systems has been investigated in offline and online environments [5], [6]. In fact, the adaptive behavior of neural networks makes them powerful tools for state observation without any *a priori* knowledge about system dynamics. In [7], a general multiple-input–multiple-output (MIMO) nonlinear system was linearized and an extended Kalman filter was used to estimate the states of the system. The gain of the proposed observer was computed by a multilayer feedforward neural network. In [8], a scheme using two separate *linear-in-parameter* neural networks (LPNN) was proposed to estimate the states of *affine single-input–single-output (SISO) nonlinear* systems. Strong assumptions, such as a strictly positive real (SPR) condition, were imposed on the output error equation. In addition, they considered *scalar* valued nonlinear functions in system dynamics. This implies that the same nonlinear terms correspond to all state variables. In [9], a general nonlinear model was considered. It was claimed that every general nonlinear model can be described by an affine model plus a bounded unmodeled dynamic term. Hence, the affine model was used for observer design. No clear method was suggested for decreasing the amount of error arbitrarily. In [10], [11], an observer for a general MIMO nonlinear system using a *linear -in-parameter* neural network was proposed and the SPR assumption was also relaxed. According to the authors, however, choosing proper values of the design parameters such as various gains and functional links of the neural networks is extremely difficult. Moreover, the observer has an open-loop structure. A nonlinear observer based on a *nonlinear-in-parameters* neural network (NLPNN) was proposed in [12]. However, a linear approximation was used using Taylor series expansion to facilitate the stability analysis. In [13], a neural network observer for continuous-time autonomous (no input) nonlinear system was proposed which represent a narrow class of nonlinear systems. Battilotti *et al.* [14] presented an adaptive output feedback controller for a class of uncertain stochastic nonlinear systems. The nonlinear part of the systems is decomposed into a part, obtained as the best approximation given by a neural network, plus an uncertain term. A LPNN was used for function approximation whose weights were tuned adaptively by the Lyapunov design method. An observer-based neural adaptive controller for SISO affine nonlinear system was developed in

Manuscript received March 8, 2004; revised May 11, 2005. This research was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada under Grant RGPIN-1345.

F. Abdollahi was with the Electrical Engineering Department, Amirkabir University of Technology, Tehran, Iran. She is now with the Department of Electrical and Computer Engineering, Concordia University, Montreal, QC H3G 2W1 Canada (e-mail: f-abdoll@ece.concordia.ca).

H. A. Talebi is with the Department of Electrical Engineering, Amirkabir University of Technology, Tehran 15914, Iran and also with the Department of Electrical and Computer Engineering, University of Western Ontario, London, ON N6A 5B9 Canada (e-mail: alit@aut.ac.ir).

R. V. Patel is with the Department of Electrical and Computer Engineering, University of Western Ontario, London, ON N6A 5B9 Canada (e-mail: rajni@eng.uwo.ca).

Digital Object Identifier 10.1109/TNN.2005.863458

1045-9227/$20.00 © 2006 IEEE

[15]. A radial basis function (RBF) type of neural network was employed to approximate the nonlinearities. In contrast to other works, in [16], the weight updating mechanism was based on the steepest descent method. The observer was based on a general model of MIMO nonlinear systems and was shown to be experimentally stable, but no mathematical proof was given to support the experiments. In [17], a recurrent neural network for general MIMO nonlinear systems was proposed. The neural network weights were updated based on the backpropagation (BP) algorithm. The SPR assumption was also relaxed. Although the stability of the observer was shown by Lyapunov's method, like most proposed neural network observers, it was linearly parameterized. This assumption greatly simplifies the analysis, but is a strong constraint since not all nonlinear functions can be represented by such equations.

In this paper, a neural network-based adaptive observer for a general model of MIMO nonlinear systems is proposed. The neural network is *nonlinear* in its parameters and can be applied to many systems with arbitrary degrees of nonlinearity and complexity. The online weight updating mechanism is a modified version of the BP algorithm with a simple structure together with an e-modification term added for robustness. The SPR assumption imposed on the output error equation is also relaxed. The paper is organized as follows. In Section II, some preliminary definitions are given. The proposed neural network is introduced in Section III. In Section IV, a mathematical proof of stability, the main part of the paper, is given. Section V gives a model of flexible-joint manipulators. The observer performance is demonstrated in Section VI by simulation carried out on single- and two-link flexible-joint manipulators.

II. NOTATIONS

The norm of a vector $x \in R^n$ and the spectral norm of a matrix $A \in R^{m\times n}$ are denoted as

$$\|x\| = \sqrt{x^T x} \quad \|A\|_s = \sqrt{\lambda_{\max}[A^T A]}$$

where $\lambda_{\max}[\cdot]$ denotes the largest eigenvalue of the positive–definite or positive–semidefinite matrix $[\cdot]$. We denote the smallest eigenvalue of a positive–definite matrix $[\cdot]$ by $\lambda_{\min}[\cdot]$. Given $A = [a_{ij}]$ and $B \in R^{m\times n}$, the Frobenius norm is defined by

$$\|A\|_F^2 = tr(A^T A) = \Sigma a_{ij}^2$$

where $tr(\cdot)$ denotes the trace of $(\cdot)$. The associated inner product is

$$\langle A, B\rangle_F = tr(A^T B).$$

The space of an L_∞ bounded signal can be defined as

$$x(t) \in L_\infty \text{ if ess } \sup_t |x(t)| < \infty.$$

Now let $x \in L_\infty$, the L_∞ norm of the signal $x(t)$ is defined as

$$\|x\|_{L_\infty} = \text{ess } \sup_t |x(t)|$$

III. THE PROPOSED NEURO-ADAPTIVE OBSERVER

Consider the general model of a nonlinear MIMO system

$$\begin{aligned} \dot{x}(t) &= f(x,u) \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

where $u \in R^{m_u}$ is the input, $y \in R^{m_y}$ is the output, $x \in R^n$ is the state vector of the system, and f is a vector-valued unknown nonlinear function. It is assumed that the nonlinear system (1) is observable. Another reasonable assumption made here is that the open-loop system is stable. In other words, the states of the system $(x(t))$ are bounded in L_∞. This is a common assumption in identification schemes.

Now, by adding and subtracting Ax, (1) becomes

$$\begin{aligned} \dot{x}(t) &= Ax + g(x,u) \\ y(t) &= Cx(t) \end{aligned} \tag{2}$$

where A is a Hurwitz matrix, the pair (C, A) is observable, and $g(x,u) = f(x,u) - Ax$.

Now, the observer model can be selected as

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x} + \hat{g}(\hat{x},u) + G(y - C\hat{x}) \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{3}$$

where $\hat{x}$ denotes the state of the observer, and the observer gain $G \in R^{n\times m_y}$ is selected such that $A - GC$ is a Hurwitz matrix. Such a gain is guaranteed to exist, since A can be selected such that (C, A) is observable. The structure of the observer is shown in Fig. 1. In this figure, $\hat{x}$ denotes the state of the recurrent model (3). Corresponding to the Hurwitz matrix A, $M(s) := (sI - A)^{-1}$ is also shown which is an $n\times n$ matrix whose elements are stable transfer functions. The key to designing a neuro-observer is to employ a neural network to identify the nonlinearity and a conventional observer to estimate the states. It is well known that a three-layer neural network is capable of approximating nonlinear systems with any degree of nonlinearity. In fact, it has been shown by many researchers (e.g., [18] and [19]) that for x restricted to a compact set S of $x \in R^n$ and for some sufficiently large number of hidden layer neurons, there exist weights and thresholds such that any continuous function on the compact set S can be represented as

$$g(x,u) = W\sigma(V\bar{x}) + \epsilon(x)$$

where W and V are the weight matrices of the output and hidden layers, respectively, $\bar{x} = [x \;\; u]$, $\epsilon(x)$ is the bounded neural network approximation error, and $\sigma(\cdot)$ is the transfer function of the hidden neurons that is usually considered as a sigmoidal function

$$\sigma_i(V_i\bar{x}) = \frac{2}{1 + \exp^{-2V_i\bar{x}}} - 1$$

where V_i is the ith row of V, and $\sigma_i(V_i\bar{x})$ is the ith element of $\sigma(V\bar{x})$.

We assume that the upper bound on fixed ideal weights W and V exist such that

$$\|W\|_F \le W_M \tag{4}$$
$$\|V\|_F \le V_M. \tag{5}$$

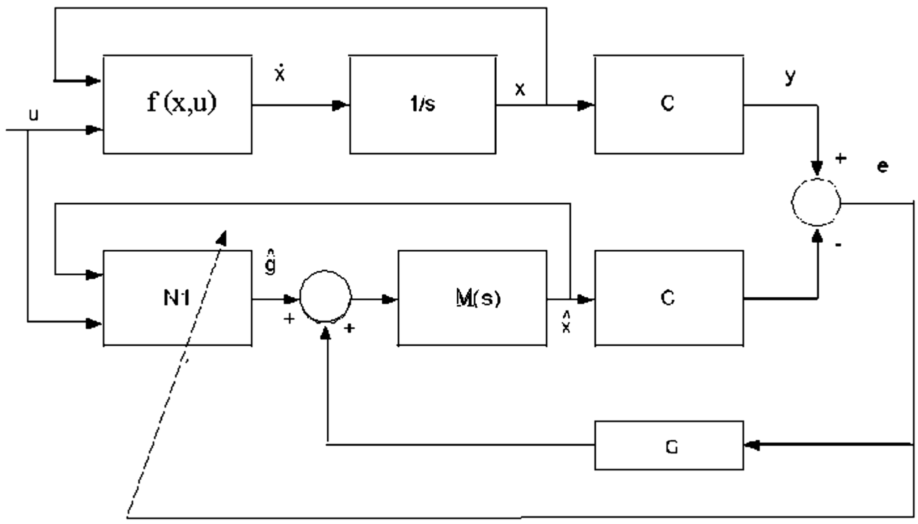

Fig. 1. Structure of the proposed neural network observer.

It is also known that the sigmoidal function is bounded by

$$\|\sigma(V\bar{x})\| \leq \sigma_m. \tag{6}$$

Thus, the function g can be approximated by

$$\hat{g}(\hat{x}, u) = \hat{W}\sigma(\hat{V}\hat{\bar{x}}). \tag{7}$$

The proposed observer is then given by

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x} + \hat{W}\sigma(\hat{V}\hat{\bar{x}}) + G(y - C\hat{x}) \\ \hat{y}(t) &= C\hat{x}(t). \end{aligned} \tag{8}$$

Defining the state estimation error as $\tilde{x} = x - \hat{x}$ and using (2), (7) and (8), we can express the error dynamics as

$$\begin{aligned} \dot{\tilde{x}}(t) &= Ax + W\sigma(V\bar{x}) - A\hat{x} - \hat{W}\sigma(\hat{V}\hat{\bar{x}}) \\ &\quad - G(Cx - C\hat{x}) + \epsilon(x) \\ \tilde{y}(t) &= C\tilde{x}(t). \end{aligned} \tag{9}$$

Now by adding $W\sigma(\hat{V}\hat{\bar{x}})$ to and subtracting from (9), we can write

$$\begin{aligned} \dot{\tilde{x}}(t) &= A_c\tilde{x} + \tilde{W}\sigma(\hat{V}\hat{\bar{x}}) + w(t) \\ \tilde{y}(t) &= C\tilde{x}(t) \end{aligned} \tag{10}$$

where $\tilde{W} = W - \hat{W}$, $A_c = A - GC$, $w(t) = W[\sigma(V\bar{x}) - \sigma(\hat{V}\hat{\bar{x}})] + \epsilon(x)$ is a bounded disturbance term i.e, $\|w(t)\| \leq \bar{w}$ for some positive constant $\bar{w}$, due to the sigmoidal function and the boundedness of the ideal neural network weights (V, W).

IV. Stability Analysis

Once the structure of the neural network is known, a proper learning rule should be defined to train the network. This weight updating mechanism is usually defined in such a way that the stability of the observer is guaranteed. Furthermore, the adaptive law should not be complicated or limited by some strong constraints. BP is one of the most popular algorithms that has been widely used for classification, recognition, identification, observation, and control problems. BP owes its popularity to the simplicity in structure which makes it a viable choice for practical problems. However, the main drawback of the previous work (e.g., [20] and [21]) is the lack of a mathematical proof of stability. In this section, we propose a new learning rule and then prove the stability of the proposed scheme by Lyapunov's direct method. The weight updating mechanism is based on the modified BP algorithm plus an e-modification term to guarantee its robustness. The following theorem gives the main result of the paper.

Theorem 1: Consider the plant model (1) and the observer model (8). If the weights of the NLPNN are updated according to

$$\dot{\hat{W}} = -\eta_1 \left(\tilde{y}^T C A_c^{-1}\right)^T (\sigma(\hat{V}\hat{\bar{x}}))^T - \rho_1\|\tilde{y}\|\hat{W} \tag{11}$$

$$\begin{aligned} \dot{\hat{V}} &= -\eta_2 \left(\tilde{y}^T C A_c^{-1}\hat{W}(I - \Lambda(\hat{V}\hat{\bar{x}}))\right)^T \operatorname{sgn}(\hat{\bar{x}})^T \\ &\quad - \rho_2\|\tilde{y}\|\hat{V} \end{aligned} \tag{12}$$

where $\Lambda(\hat{V}\hat{\bar{x}}) = \operatorname{diag}\left\{\sigma_i^2(\hat{V}_i\hat{\bar{x}})\right\}$, $i = 1, 2, \ldots, m$ and $\operatorname{sgn}(\hat{\bar{x}})$ is the sign function

$$\operatorname{sgn}(\hat{\bar{x}}) = \begin{cases} 1, & \text{for } \hat{\bar{x}} > 0 \\ 0, & \text{for } \hat{\bar{x}} = 0 \\ -1, & \text{for } \hat{\bar{x}} < 0 \end{cases}$$

then $\tilde{x}, \tilde{W}, \tilde{V}, \tilde{y} \in L_\infty$, i.e., the estimation error, weights error, and the output error are all bounded. In these equations, η_1, $\eta_2 > 0$ are the learning rates, $J = (1/2)(\tilde{y}^T\tilde{y})$ is the objective function and ρ_1, ρ_2 are small positive numbers.

In what follows, we first show that the first terms in (11) and (12) are the BP terms and the second terms are the e-modification terms for incorporating damping in the equations, i.e.,

$$\dot{\hat{W}} = -\eta_1 \left(\frac{\partial J}{\partial \hat{W}}\right) - \rho_1\|\tilde{y}\|\hat{W} \tag{13}$$

$$\dot{\hat{V}} = -\eta_2 \left(\frac{\partial J}{\partial \hat{V}}\right) - \rho_2\|\tilde{y}\|\hat{V}. \tag{14}$$

Proof: Let us define

$$\text{net}_{\hat{v}} = \hat{V}\hat{\bar{x}} \tag{15}$$

$$\text{net}_{\hat{w}} = \hat{W}\sigma(\hat{V}\hat{\bar{x}}). \tag{16}$$

Therefore, $(\partial J/\partial \hat{W})$ and $(\partial J/\partial \hat{V})$ can be computed as [22]

$$\begin{aligned}\frac{\partial J}{\partial \hat{W}} &= \frac{\partial J}{\partial \mathrm{net}_{\hat{w}}} \cdot \frac{\partial \mathrm{net}_{\hat{w}}}{\partial \hat{W}} \\ \frac{\partial J}{\partial \hat{V}} &= \frac{\partial J}{\partial \mathrm{net}_{\hat{v}}} \cdot \frac{\partial \mathrm{net}_{\hat{v}}}{\partial \hat{V}}.\end{aligned} \tag{17}$$

We have

$$\begin{aligned}\frac{\partial J}{\partial \mathrm{net}_{\hat{w}}} &= \frac{\partial J}{\partial \tilde{y}} \cdot \frac{\partial \tilde{y}}{\partial \hat{x}} \cdot \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{w}}} = -\tilde{y}^T C \cdot \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{w}}} \\ &= -\tilde{x}^T C^T C \cdot \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{w}}} \\ \frac{\partial J}{\partial \mathrm{net}_{\hat{v}}} &= \frac{\partial J}{\partial \tilde{y}} \cdot \frac{\partial \tilde{y}}{\partial \hat{x}} \cdot \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{v}}} = -\tilde{y}^T C \cdot \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{v}}} \\ &= -\tilde{x}^T C^T C \cdot \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{v}}}\end{aligned} \tag{18}$$

and

$$\begin{aligned}\frac{\partial \mathrm{net}_{\hat{w}}}{\partial \hat{W}} &= \sigma(\hat{V}\hat{\bar{x}}) \\ \frac{\partial \mathrm{net}_{\hat{v}}}{\partial \hat{V}} &= \hat{\bar{x}}.\end{aligned} \tag{19}$$

Now, by using (8) and the definitions of $\mathrm{net}_{\hat{v}}$ and $\mathrm{net}_{\hat{w}}$ as (15) and (16), respectively, we can write

$$\begin{aligned}\frac{\partial \dot{\hat{x}}(t)}{\partial \mathrm{net}_{\hat{w}}} &= A_c \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{w}}} + I \\ \frac{\partial \dot{\hat{x}}(t)}{\partial \mathrm{net}_{\hat{v}}} &= A_c \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{v}}} + \hat{W}(I - \Lambda(\hat{V}\hat{\bar{x}})).\end{aligned} \tag{20}$$

The above equations represent a set of nonlinear dynamical systems and the so-called backpropagation in time (dynamic backpropagation) should be used to solve for the gradients $(\partial \hat{x}/\partial \mathrm{net}_{\hat{w}})$ and $(\partial \hat{x}/\partial \mathrm{net}_{\hat{v}})$. However, this adds to the complexity of the observer and makes the real-time implementation of the approach very difficult. We suggest to use the static approximation of the gradients, i.e., by setting $\dot{\hat{x}} = 0$ in (20), we have

$$\begin{aligned}\frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{w}}} &\approx -A_c^{-1} \\ \frac{\partial \hat{x}}{\partial \mathrm{net}_{\hat{v}}} &\approx -A_c^{-1}\hat{W}(I - \Lambda(\hat{V}\hat{\bar{x}})).\end{aligned} \tag{21}$$

Now, by using (18), (19), and (21), the learning rules (13) and (14) can be written as

$$\begin{aligned}\dot{\hat{W}} &= -\eta_1 \left(\tilde{x}^T C^T C A_c^{-1}\right)^T (\sigma(\hat{V}\hat{\bar{x}}))^T \\ &\quad - \rho_1 \|C\tilde{x}\|\hat{W}\end{aligned} \tag{22}$$

$$\begin{aligned}\dot{\hat{V}} &= -\eta_2 \left(\tilde{x}^T C^T C A_c^{-1}\hat{W}(I - \Lambda(\hat{V}\hat{\bar{x}}))\right)^T \hat{\bar{x}}^T \\ &\quad - \rho_2 \|C\tilde{x}\|\hat{V}.\end{aligned} \tag{23}$$

Now, the learning rules (22) and (23) in terms of the weight errors $\tilde{W} = W - \hat{W}$ and $\tilde{V} = V - \hat{V}$, can be written as

$$\dot{\tilde{W}} = \eta_1 \left(\tilde{x}^T C^T C A_c^{-1}\right)^T (\sigma(\hat{V}\hat{\bar{x}}))^T + \rho_1 \|C\tilde{x}\|\hat{W} \tag{24}$$

$$\begin{aligned}\dot{\tilde{V}} &= \eta_2 \left(\tilde{x}^T C^T C A_c^{-1}\hat{W}(I - \Lambda(\hat{V}\hat{\bar{x}}))\right)^T \hat{\bar{x}}^T \\ &\quad + \rho_2 \|C\tilde{x}\|\hat{V}.\end{aligned} \tag{25}$$

In order to simplify the stability analysis, we replace $\hat{\bar{x}}$ by $\mathrm{sgn}(\hat{\bar{x}})$ in the above equation

$$\begin{aligned}\dot{\tilde{V}} &= \eta_2 \left(\tilde{x}^T C^T C A_c^{-1}\hat{W}(I - \Lambda(\hat{V}\hat{\bar{x}}))\right)^T \mathrm{sgn}(\hat{\bar{x}})^T \\ &\quad + \rho_2 \|C\tilde{x}\|\hat{V}.\end{aligned} \tag{26}$$

As will be clear later, this modification is necessary to derive (33), since $\mathrm{sgn}(\hat{\bar{x}})$ is bounded but this is not necessarily true for $\hat{\bar{x}}$. Note that by using the sign of $\hat{\bar{x}}$, the weight update is guaranteed to move in the right direction.

It can be seen that the learning rule (24) and (26) are equivalent to (11) and (12) expressed in terms of $\tilde{W}$ and $\tilde{V}$. We now consider the positive definite Lyapunov function candidate

$$L = \frac{1}{2}\tilde{x}^T P \tilde{x} + \frac{1}{2} tr(\tilde{W}^T \tilde{W}) + \frac{1}{2} tr(\tilde{V}^T \tilde{V}) \tag{27}$$

where $P = P^T$ is a positive–definite matrix satisfying

$$A_c^T P + P A_c = -Q \tag{28}$$

for the Hurwitz matrix A_c and some positive–definite matrix Q. The time derivative of (27) is given by

$$\dot{L} = \frac{1}{2}\dot{\tilde{x}}^T P \tilde{x} + \frac{1}{2}\tilde{x}^T P \dot{\tilde{x}} + tr(\tilde{W}^T \dot{\tilde{W}}) + tr(\tilde{V}^T \dot{\tilde{V}}). \tag{29}$$

Now, by substituting (10), (24), (26) and (28) into (29), one can get

$$\begin{aligned}\dot{L} &= -\frac{1}{2}\tilde{x}^T Q \tilde{x} + \tilde{x}^T P(\tilde{W}\sigma(\hat{V}\hat{\bar{x}}) + w) \\ &\quad + tr\left(\tilde{W}^T l_1 \tilde{x}\sigma(\hat{V}\hat{\bar{x}})^T + \tilde{W}^T \rho_1 \|C\tilde{x}\|(W - \tilde{W})\right) \\ &\quad + tr\left(\tilde{V}^T (I - \Lambda(\hat{V}\hat{\bar{x}}))^T \hat{W}^T l_2 \tilde{x}\mathrm{sgn}(\hat{\bar{x}})^T \right. \\ &\qquad \left. + \tilde{V}^T \rho_2 \|C\tilde{x}\|(V - \tilde{V})\right)\end{aligned} \tag{30}$$

where $l_1 = \eta_1 A_c^{-T} C^T C$, $l_2 = \eta_2 A_c^{-T} C^T C$. Next, we use the inequalities

$$\begin{aligned}tr(\tilde{W}^T(W - \tilde{W})) &\leq W_M\|\tilde{W}\| - \|\tilde{W}\|^2 \\ tr(\tilde{V}^T(V - \tilde{V})) &\leq V_M\|\tilde{V}\| - \|\tilde{V}\|^2 \\ tr\left(\tilde{W}^T l_1 \tilde{x}\sigma(\hat{V}\hat{\bar{x}})^T\right) &\leq \sigma_m \|\tilde{W}^T\|\|l_1\|\|\tilde{x}\|\end{aligned} \tag{31}$$

where W_M, V_M and σ_m are given by (4)–(6). Note that the last inequality in (31) is obtained using the fact that, for two column vectors A and B, the following holds:

$$tr(AB^T) = B^T A. \tag{32}$$

Now, by using $\|\hat{W}\| = \|W - \tilde{W}\| \leq W_M + \|\tilde{W}\|$, $1 - \sigma_m^2 \leq 1$, and (32), the following inequality is obtained:

$$\begin{aligned}&tr\left(\tilde{V}^T (I - \Lambda(\hat{V}\hat{\bar{x}}))^T \hat{W}^T l_2 \tilde{x}\mathrm{sgn}(\hat{\bar{x}})^T\right) \\ &\qquad \leq \|\tilde{V}\|(W_M + \|\tilde{W}\|)\|l_2\|\|\tilde{x}\|.\end{aligned} \tag{33}$$

Then by using (31) and (33), we get

$$\begin{aligned}\dot{L} \leq& -\frac{1}{2}\lambda_{\min}(Q)\|\tilde{x}\|^2 + \|\tilde{x}\|\|P\|(\|\tilde{W}\|\sigma_m + \bar{w})\\ &+ \sigma_m\|\tilde{W}\|\|l_1\|\|\tilde{x}\| + \left(W_M\|\tilde{W}\| - \|\tilde{W}\|^2\right)\rho_1\|C\|\|\tilde{x}\|\\ &+ \|\tilde{V}\|\|l_2\|(W_M + \|\tilde{W}\|)\|\tilde{x}\|\\ &+ \rho_2\|C\|\|\tilde{x}\|\left(V_M\|\tilde{V}\| - \|\tilde{V}\|^2\right)\\ =& F. \end{aligned} \tag{34}$$

By completing the squares for the terms involving $\|\tilde{W}\|$ and $\|\tilde{V}\|$, we try to find some conditions on $\|\tilde{x}\|$, independent of the neural network weights error, that make the derivative of the Lyapunov function candidate negative. First, by defining $K_1 = (\|l_2\|/2)$ and by adding and subtracting $K_1^2\|\tilde{W}\|^2\|\tilde{x}\|$ and $\|\tilde{V}\|^2\|\tilde{x}\|$ to the right hand side of (34), we can get

$$\begin{aligned}F =& -\frac{1}{2}\lambda_{\min}(Q)\|\tilde{x}\|^2\\ &+ \Big[\|P\|\bar{w} - \left(\rho_1\|C\| - K_1^2\right)\|\tilde{W}\|^2\\ &\quad + (\|P\|\sigma_m + \sigma_m\|l_1\| + \rho_1\|C\|W_M)\|\tilde{W}\|\\ &\quad + (\rho_2\|C\|V_M + \|l_2\|W_M)\|\tilde{V}\|\\ &\quad - (\rho_2\|C\| - 1)\|\tilde{V}\|^2 - (K_1\|\tilde{W}\| - \|\tilde{V}\|)^2\Big]\|\tilde{x}\|. \end{aligned} \tag{35}$$

Next, let us introduce K_2 and K_3 as

$$K_2 = \frac{\rho_1 W_M\|C\| + \sigma_m\|l_1\| + \|P\|\sigma_m}{2\left(\rho_1\|C\| - K_1^2\right)}$$
$$K_3 = \frac{\rho_2\|C\|V_M + \|l_2\|W_M}{2\left(\rho_2\|C\| - 1\right)}.$$

Then, $K_2^2\|\tilde{x}\|$ and $K_3^2\|\tilde{x}\|$ are added to and subtracted from (35)

$$\begin{aligned}F =& -\frac{1}{2}\lambda_{\min}(Q)\|\tilde{x}\|^2\\ &+ \Big[\|P\|\bar{w} + \left(\rho_1\|C\| - K_1^2\right)K_2^2 + (\rho_2\|C\| - 1)K_3^2\\ &\quad - \left(\rho_1\|C\| - K_1^2\right)(K_2 - \|\tilde{W}\|)^2 - (\rho_2\|C\| - 1)\\ &\quad \times (K_3 - \|\tilde{V}\|)^2 - (K_1\|\tilde{W}\| - \|\tilde{V}\|)^2\Big]\|\tilde{x}\|. \end{aligned} \tag{36}$$

Now assuming $\rho_1 \geq \left(K_1^2/\|C\|\right)$, $\rho_2 \geq (1/\|C\|)$ and considering that the last three terms of (37) are negative, we get

$$\begin{aligned}F \leq& -\frac{1}{2}\lambda_{\min}(Q)\|\tilde{x}\|^2 + \|\tilde{x}\|\\ &\times \left(\|P\|\bar{w} + \left(\rho_1\|C\| - K_1^2\right)K_2^2 + (\rho_2\|C\| - 1)K_3^2\right). \end{aligned} \tag{37}$$

Therefore, the following condition on $\|\tilde{x}\|$ guarantees the negative semi-definiteness of $\dot{L}$

$$\begin{aligned}\|\tilde{x}\| >& \frac{2\left(\|P\|\bar{w} + \left(\rho_1\|C\| - K_1^2\right)K_2^2 + (\rho_2\|C\| - 1)K_3^2\right)}{\lambda_{\min}(Q)}\\ =& b. \end{aligned} \tag{38}$$

In fact, $\dot{L}$ is negative definite outside the ball with radius b described as $\chi = \{\tilde{x} \mid \|\tilde{x}\| > b\}$, and $\tilde{x}$ is uniformly ultimately bounded. The region inside the ball is attractive, since the increase of $\dot{L}$ for smaller values of $\|\tilde{x}\|$ will increase L and $\tilde{x}$, which brings the $\tilde{x}$ outside the ball χ where $\dot{L}$ is negative semi-definite and results in reducing L and $\tilde{x}$. The above analysis shows the ultimate boundedness of $\tilde{x}$. To show the boundedness of the weight errors $\tilde{W}$ and $\tilde{V}$, consider (24) and (26), which can be rewritten as

$$\begin{aligned}\dot{\tilde{W}} &= f_1(\tilde{x}, \hat{V}) + \rho_1\|C\tilde{x}\|\hat{W}\\ &= f_1(\tilde{x}, \hat{V}) + \alpha_1 W - \alpha_1\tilde{W} \end{aligned} \tag{39}$$
$$\begin{aligned}\dot{\tilde{V}} &= f_2(\tilde{x}, \hat{W}, \hat{V}) + \rho_2\|C\tilde{x}\|\hat{V}\\ &= f_2(\tilde{x}, \hat{W}, \hat{V}) + \alpha_2 V - \alpha_2\tilde{V} \end{aligned} \tag{40}$$

where

$$\begin{aligned}f_1(\tilde{x}, \hat{V}) &= \eta_1\left(\tilde{x}^T C^T C A_c^{-1}\right)^T (\sigma(\hat{V}\hat{\bar{x}}))^T\\ f_2(\tilde{x}, \hat{W}, \hat{V}) &= \eta_2\left(\tilde{x}^T C^T C A_c^{-1}\hat{W}(I - \Lambda(\hat{V}\hat{\bar{x}}))\right)^T \operatorname{sgn}(\hat{\bar{x}})^T\\ \alpha_1 &= \rho_1\|C\tilde{x}\|\\ \alpha_2 &= \rho_2\|C\tilde{x}\|. \end{aligned}$$

It can be seen that $f_1(\cdot)$ is bounded since $\tilde{x}$ and $\sigma(\hat{V}\hat{\bar{x}})$ are both bounded, C is bounded, and A_c is a Hurwitz matrix. Given the fact the ideal weight W is fixed, (39) can be regarded as a linear system with bounded input $(f_1(\tilde{x}, \hat{V}) + \alpha_1 W)$. It is clear that this system is stable since α_1 is positive and the system input remains bounded. Hence, the boundedness of $\tilde{W}$ is also ensured. Given that $\tilde{W} \in L_\infty$, it can be observed that $f_2(\cdot)$ is also bounded since all its arguments are bounded including $\Lambda(\cdot)$ as defined below (12). Consequently, similar analysis shows that (40) also represents a stable bounded input linear system and hence $\tilde{V} \in L_\infty$. The key to the previous analysis is that $\tilde{V}$ only appears in $f_1(\cdot)$ and $f_2(\cdot)$ as bounded functions ($\sigma(\cdot)$ and $\Lambda(\cdot)$).

The size of the estimation error bound b can be kept small by proper selection of the damping factors, A_c and the learning rates (through K_2 and K_3) such that higher accuracy can be achieved. It should be noted that since ρ_1 and ρ_2 are design parameters, the conditions on them do not restrict the applicability of the proposed approach.

Remark 1: In many cases, not all system states directly appear in the output of the system. Hence, some elements of C would be zero and this will slow down the learning process because of the structure of the BP algorithm (see (24) and (26)). It is suggested that for the purpose of training only, the output matrix C is modified to C_1 such that all the states appear in the

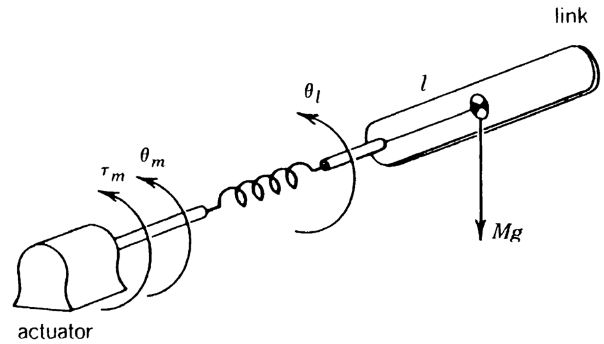

Fig. 2. Schematic of flexible-joint manipulator modeled by torsional spring.