2017 American Control Conference
Sheraton Seattle Hotel
May 24–26, 2017, Seattle, USA

Neural Network Based Repetitive Learning Control of Robot Manipulators

Necati Cobanoglu, Enver Tatlicioglu*, and Erkan Zergeroglu

Abstract— **Control of robot manipulators performing periodic tasks is considered in this work. The control problem is complicated by presence of uncertainties in the robot manipulator's dynamic model. To address this restriction, a model free repetitive learning controller design is aimed. To reduce the heavy control effort, a neural network based compensation term is fused with the repetitive learning controller. The convergence of the tracking error to the origin is ensured via Lyapunov based techniques. Numerical simulations and experiments are performed to demonstrate the viability of the proposed controller.**

I. Introduction

Several tasks performed by robot manipulators require the joints to track periodic reference positions. In performing these periodic tasks, in addition to standard robot controllers, learning controllers are also preferred. When robot manipulator's dynamic model is uncertain learning controllers have the advantage of *learning* the uncertainties of the dynamic model via an update law. More can be found about learning controllers in the surveys [1], [2] and in [3].

A good amount of past research was devoted to repetitive learning controller design and their extensions. The main usage of repetitive learning controllers is for tracking or rejecting periodic signals with known period. Some of the earlier works are [4], [5], [6], [7]. Later to ensure boundedness of the closed loop signals saturation function based update laws were proposed in [8] and [9]. In some recent works, such as [10], a linearized update law was obtained from the saturation function based by using Pade approximants. While this approach allowed utilizing linear controllers in conjunction with the learning law, asymptotic stability obtained in the previous works was lost and convergence of the tracking error to a hyperball centered at the origin (whose size can be reduced with increasing control gains) can be ensured. Some other extensions [11] and several applications [12] of repetitive learning controllers are also available.

A shortcoming of the repetitive learning controllers is the *learning* of the uncertainties in the dynamic model as a whole. This yields an increased feedback gain in the learning update rule which may cause chattering like problems at the end of the periods. To propose a solution, in this work, neural networks are considered to be used in conjunction with repetitive learning controllers having a saturation function based update rule. The main reason of preferring neural networks is their performance in learning modeling uncertainties [13], [14]. Specifically, the uncertainties in the dynamic model are combined in a vector and a one layer neural network model of this vector is considered. A novel nonlinear proportional derivative (PD) controller having a repetitive learning component and a neural network component is designed. Neural networks are considered to learn some part of the modeling uncertainties while repetitive learning controller will compensate for the rest of the modeling uncertainties. Stability of the closed loop system is investigated with Lyapunov based stability analysis methods and asymptotic stability of the joint position tracking error is ensured. To back the stability analysis up, results of simulations and experiments are presented.

II. Dynamic Model and Model Properties

The dynamic/mathematical model of an n degree of freedom revolute joint robot manipulator is given as [15]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F_d\dot{q} = \tau \tag{1}$$

where $q(t)$, $\dot{q}(t)$, $\ddot{q}(t) \in \mathbb{R}^n$ denote joint positions, velocities, and accelerations respectively, $M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is the centripetal Coriolis matrix, $G(q) \in \mathbb{R}^n$ represents the gravitational effects, $F_d \in \mathbb{R}^{n\times n}$ denotes the constant frictional effects, and $\tau(t) \in \mathbb{R}^n$ is the control input torque.

The dynamic model terms satisfy the standard properties given below.

Property 1: The inertia matrix is positive definite and symmetric and satisfies the following inequalities [15]

$$\underline{m}\left\|\eta\right\| \leq \eta^T M(q)\eta \leq \bar{m}\left\|\eta\right\| \;\; \forall\eta \in \mathbb{R}^n \tag{2}$$

where $\underline{m}$, $\bar{m}$ are known positive bounding constants.

Property 2: The inertia matrix and the centripetal Coriolis matrix satisfy the skew symmetry property [15]

$$\eta^T\left(\dot{M} - C\right)\eta = 0 \;\forall\eta \in \mathbb{R}^n. \tag{3}$$

Property 3: The norms of the dynamic model terms can be upper bounded as [15]

$$\left\|C(q,\dot{q})\right\|_{i\infty} \leq \zeta_c\left\|\dot{q}\right\| \;,\; \left\|G(q)\right\| \leq \zeta_g \;,\; \left\|F_d\right\|_{i\infty} \leq \zeta_f \tag{4}$$

where ζ_c, ζ_g, ζ_f are known positive bounding constants, and $\left\|\cdot\right\|_{i\infty}$ is the induced infinity norm of a matrix.

N. Cobanoglu and E. Tatlicioglu are with the Department of Electrical & Electronics Engineering, Izmir Institute of Technology, Izmir, 35430 Turkey (Phone: +90 (232) 7506536; Fax: +90 (232) 7506599; E-mail: [necaticobanoglu,envertatlicioglu]@iyte.edu.tr).
E. Zergeroglu is with the Department of Computer Engineering, Gebze Technical University, 41400, Gebze, Kocaeli, Turkey (Email: e.zerger@gtu.edu.tr).
E. Tatlicioglu is partially funded by The Scientific and Technological Research Council of Turkey via grant number 115E726.
*To whom all the correspondence should be addressed.

978-1-5090-5992-8/$31.00 ©2017 AACC

III. Error System Development and Control Design

The main objective of the control input torque design is to ensure tracking of a periodic desired joint position vector. Mathematically speaking, $q(t) \rightarrow q_d(t)$ is aimed where $q_d(t) \in \mathbb{R}^n$ is periodic with a known period T in the sense that $q_d(t) = q_d(t-T)$, $\dot{q}_d(t) = \dot{q}_d(t-T)$, $\ddot{q}_d(t) = \ddot{q}_d(t-T)$. Desired joint position vector and its time derivatives are bounded functions of time.

The control problem is constrained by the dynamic model terms in (1) being uncertain and thus are not available for control design. In view of this, the control design should be model independent.

The joint position tracking error $e(t) \in \mathbb{R}^n$ is defined as

$$e \triangleq q_d - q. \tag{5}$$

Another error, shown with $r(t) \in \mathbb{R}^n$, is defined as

$$r \triangleq \dot{e} + \alpha e \tag{6}$$

where $\alpha \in \mathbb{R}^{n\times n}$ is a constant, positive definite, diagonal control gain matrix. In an attempt to obtain open loop error system for $r(t)$, first the time derivative of (6) is taken which is then multiplied with the inertia matrix to yield

$$M\dot{r} = -Cr - \tau + \Omega \tag{7}$$

where (1) and (6) were utilized and $\Omega(q, \dot{q}, e, \dot{e}, \ddot{q}_d) \in \mathbb{R}^n$ is defined as

$$\Omega \triangleq M(q)(\ddot{q}_d + \alpha\dot{e}) + C(q, \dot{q})(\dot{q}_d + \alpha e) + G(q) + F_d\dot{q}. \tag{8}$$

An auxiliary vector, shown with $\Omega_d(q_d, \dot{q}_d, \ddot{q}_d) \in \mathbb{R}^n$, is obtained by setting $q \rightarrow q_d$ and $\dot{q} \rightarrow \dot{q}_d$ in Ω which is given as

$$\Omega_d \triangleq M(q_d)\ddot{q}_d + C(q_d, \dot{q}_d)\dot{q}_d + G(q_d) + F_d\dot{q}_d. \tag{9}$$

Property 4: Via utilizing the universal approximation property of neural networks [13], [14], [16], [17], the auxiliary vector Ω_d is considered to be written with one layer neural network as [15]

$$\Omega_d = \varphi^T\sigma + \epsilon \tag{10}$$

where $\varphi \in \mathbb{R}^{3n\times n}$ is the constant ideal weight matrix, $\sigma(x_d) \in \mathbb{R}^{3n}$ is the activation function, $\epsilon(x_d) \in \mathbb{R}^n$ is the functional reconstruction error, and $x_d(t) \triangleq \begin{bmatrix} q_d^T & \dot{q}_d^T & \ddot{q}_d^T \end{bmatrix}^T \in \mathbb{R}^{3n}$ is the combined form of desired joint position and its time derivatives. For feedback control using neural networks, usually the activation function is required to be smooth enough so that at least its first time derivative exists. To meet this requirement, in this work, hyperbolic tangent function is preferred as the activation function. The entries of the functional reconstruction error are bounded in the sense that [13]

$$\bar{\epsilon}_i \geq |\epsilon_i(x_d)| \;\; \forall i = 1, \cdots, n \tag{11}$$

where $\bar{\epsilon}_i$ are constant, positive bounding scalars. Furthermore, since the functional reconstruction error is a function of only the desired joint position and its time derivatives, it is also periodic with period T.

An auxiliary error–like term, shown with $\chi(t) \in \mathbb{R}^n$, is defined as

$$\chi \triangleq \Omega - \Omega_d. \tag{12}$$

It is remarked that, as shown in [8], the norm of χ can be upper bounded as

$$\|\chi\| \leq \rho(\|e\|)\|z\| \tag{13}$$

where $\rho(\|e\|) \in \mathbb{R}$ is a known, positive bounding function and $z(t) \triangleq \begin{bmatrix} e^T & r^T \end{bmatrix}^T \in \mathbb{R}^{2n}$ is the combined error vector.

In view of (10) and (12), (7) can be rewritten as

$$M\dot{r} = -Cr - \tau + \varphi^T\sigma + \epsilon + \chi. \tag{14}$$

Based on the open loop error system in (14), the control input torque is designed as

$$\tau = K_r r + k_n\rho^2 r + e + \hat{\epsilon}(t) + \hat{\Omega}(t) \tag{15}$$

where $K_r \in \mathbb{R}^{n\times n}$ is a constant, positive definite, diagonal control gain matrix, $k_n \in \mathbb{R}$ is a constant, positive scalar damping gain. In (15), $\hat{\epsilon}(t) \in \mathbb{R}^n$ is the learning component of the control input torque that is updated according to

$$\hat{\epsilon}(t) = \mathrm{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T)) + k_l r \tag{16}$$

where $k_l \in \mathbb{R}$ is a constant, positive scalar control gain, Sat is the vector form of the standard saturation function where $\bar{\epsilon} \triangleq \begin{bmatrix} \bar{\epsilon}_1 & \cdots & \bar{\epsilon}_n \end{bmatrix}^T$ denotes its limits. Also in (15), $\hat{\Omega}(t) \in \mathbb{R}^n$ is the neural network component of the control input torque that is generated as

$$\hat{\Omega} = \hat{\varphi}^T\sigma \tag{17}$$

where $\hat{\varphi}(t) \in \mathbb{R}^{3n\times n}$ is the estimated weight matrix generated online according to

$$\dot{\hat{\varphi}} = k_{nn}\sigma r^T \tag{18}$$

where $k_{nn} \in \mathbb{R}$ is a constant, positive scalar control gain. When boundedness of the entries of $\hat{\varphi}(t)$ is a concern, a projection algorithm (such as the one in [18]) can be utilized on the right hand side of (18) to keep them remain between some lower and upper bounds.

From the control input torque in (15) and its components in (16) and (17), it is clear that knowledge of dynamic model terms in (1) is not required by the control input torque design. The control input torque in (15) can be considered as a nonlinear PD controller with repetitive learning and neural network components. Each term of the control input torque in (15) are now discussed briefly. The term $K_r r$ is a feedback term, the term $k_n\rho^2 r$ is introduced to damp out the negative effects of χ in the open loop error system, the term e will cancel a cross term that will appear in the Lyapunov type stability analysis, the term $\hat{\Omega}(t)$ is introduced to update the neural network weight matrix, while $\hat{\epsilon}(t)$ will compensate for the negative effects of the functional reconstruction error.

Substituting the control input torque in (15) into the open loop error system in (14) gives the following closed loop error system

$$M\dot{r} = -Cr - K_r r - e + \chi - k_n \rho^2 r + \epsilon - \hat{\epsilon} + \tilde{\varphi}^T \sigma \quad (19)$$

where $\tilde{\varphi}(t) \in \mathbb{R}^{3n\times n}$ is the difference between the ideal weight matrix and the estimated weight matrix

$$\tilde{\varphi} \triangleq \varphi - \hat{\varphi}. \quad (20)$$

IV. Stability analysis

Now the stability of the closed loop error system will be investigated. Lyapunov based stability analysis method will be preferred. Following theorem is introduced.

Theorem 1: Let $V(t) \in \mathbb{R}$ be defined as

$$\begin{aligned} V &\triangleq \frac{1}{2}e^T e + \frac{1}{2}r^T M r + \frac{1}{2k_{nn}}\text{tr}\{\tilde{\varphi}^T\tilde{\varphi}\} \\ &\quad + \frac{1}{2k_l}\int_{t-T}^{t} \left\|\text{Sat}_{\bar{\epsilon}}(\epsilon(\nu)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(\nu))\right\|^2 d\nu \quad (21) \end{aligned}$$

where $\text{tr}\{\cdot\}$ is the trace operator. From the above definition, $V(t)$ is non–negative. Taking the time derivative of the Lyapunov function gives

$$\begin{aligned} \dot{V} &= e^T\dot{e} + r^T M\dot{r} + \frac{1}{2}r^T\dot{M}r + \frac{1}{k_{nn}}\text{tr}\{\tilde{\varphi}^T\dot{\tilde{\varphi}}\} \\ &\quad + \frac{1}{2k_l}\left\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\right\|^2 \\ &\quad - \frac{1}{2k_l}\left\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t-T)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T))\right\|^2 \quad (22) \end{aligned}$$

where Leibniz rule [19] was utilized. Substitutions from the error system will be done subsequently. Before that the term in the last line of the time derivative of the Lyapunov function is examined

$$\begin{aligned} &\text{Sat}_{\bar{\epsilon}}(\epsilon(t-T)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T)) \\ &= \text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \hat{\epsilon}(t) + k_l r \\ &= \epsilon(t) - \hat{\epsilon}(t) + k_l r \quad (23) \end{aligned}$$

where for the first equality the periodicity of $\epsilon(t)$ and (16) were utilized, while the boundedness of the entries of $\epsilon(t)$ in (11) yielded the second equality. Substituting (6) for $\dot{e}$, (19) for $M\dot{r}$, (18) along with the time derivative of (20) for $\dot{\tilde{\varphi}}$, and (23) into (22) is found as

$$\begin{aligned} \dot{V} &= e^T(r - \alpha e) + r^T\left(-Cr - K_r r - e + \chi - k_n\rho^2 r \right. \\ &\quad \left. + \epsilon - \hat{\epsilon} + \tilde{\varphi}^T\sigma\right) + \frac{1}{2}r^T\dot{M}r - \text{tr}\{\tilde{\varphi}^T\sigma r^T\} \\ &\quad + \frac{1}{2k_l}\left\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\right\|^2 \\ &\quad - \frac{1}{2k_l}\left\|\epsilon(t) - \hat{\epsilon}(t) + k_l r\right\|^2. \quad (24) \end{aligned}$$

Following property of the trace operator is essential [15]

$$\text{tr}\{\tilde{\varphi}^T\sigma r^T\} = r^T\tilde{\varphi}^T\sigma. \quad (25)$$

From (24), canceling out common terms, utilizing skew symmetry property in (3), substituting (25), rewriting the last term, and then regrouping results in

$$\begin{aligned} \dot{V} &= -e^T\alpha e - r^T K_r r + r^T\left(\chi - k_n\rho^2 r\right) + r^T(\epsilon - \hat{\epsilon}) \\ &\quad + \frac{1}{2k_l}\left\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\right\|^2 \\ &\quad - \frac{1}{2k_l}\left\|\epsilon(t) - \hat{\epsilon}(t)\right\|^2 - (\epsilon(t) - \hat{\epsilon}(t))^T r \\ &\quad - \frac{k_l}{2}\|r\|^2 \quad (26) \end{aligned}$$

from which canceling out common terms gives

$$\begin{aligned} \dot{V} &= -e^T\alpha e - r^T K_r r - \frac{k_l}{2}\|r\|^2 + r^T\left(\chi - k_n\rho^2 r\right) \\ &\quad + \frac{1}{2k_l}\left\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\right\|^2 \\ &\quad - \frac{1}{2k_l}\left\|\epsilon(t) - \hat{\epsilon}(t)\right\|^2. \quad (27) \end{aligned}$$

Utilizing (13) with the bracketed term in the first line gives

$$\begin{aligned} r^T\left(\chi - k_n\rho^2 r\right) &\leq \rho\|r\|\|z\| - k_n\rho^2\|r\|^2 \\ &= -\left(\sqrt{k_n}\rho\|r\| - \frac{1}{2\sqrt{k_n}}\|z\|\right)^2 \\ &\quad + \frac{1}{4k_n}\|z\|^2 \\ &\leq \frac{1}{4k_n}\|z\|^2. \quad (28) \end{aligned}$$

If $\|\epsilon(t) - \hat{\epsilon}(t)\|^2$ can be proven to be greater than $\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2$ then asymptotic stability can be achieved. From [9], it is clear that

$$\|\epsilon(t) - \hat{\epsilon}(t)\|^2 \geq \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2. \quad (29)$$

Combining (28) and (29) with (27) gives

$$\begin{aligned} \dot{V} &\leq -e^T\alpha e - r^T K_r r - \frac{k_l}{2}\|r\|^2 + \frac{1}{4k_n}\|z\|^2 \\ &\leq -\left[\min\{\lambda_{\min}(\alpha), \lambda_{\min}(K_r) + \frac{k_l}{2}\} - \frac{1}{4k_n}\right]\|z\|^2 \\ &= -\beta\|z\|^2 \quad (30) \end{aligned}$$

with positive $\beta \triangleq \left[\min\{\lambda_{\min}(\alpha), \lambda_{\min}(K_r) + \frac{k_l}{2}\} - \frac{1}{4k_n}\right]$ where $\lambda_{\min}(\alpha)$ and $\lambda_{\min}(K_r)$ denote the minimum eigenvalue of α and K_r, respectively. From the structures of (21) and (30), $V(t) \in \mathcal{L}_\infty$ can be concluded. And from (21), $e(t)$, $r(t) \in \mathcal{L}_\infty$ follows. From (6), $\dot{e}(t) \in \mathcal{L}_\infty$ can be shown. From (5) and its time derivative, $q(t)$, $\dot{q}(t) \in \mathcal{L}_\infty$ are proven. Utilizing the above boundedness statements along with the boundedness of the desired trajectory and its time derivatives, from (8), $\Omega(t) \in \mathcal{L}_\infty$ can be proven. Since the output of the saturation function is always bounded then from (16), $\hat{\epsilon}(t) \in \mathcal{L}_\infty$ is ensured. Provided that $\hat{\varphi}(t) \in \mathcal{L}_\infty$ (which can be ensured via utilizing a bounding projection algorithm as in [18]), from (17), $\hat{\Omega}(t) \in \mathcal{L}_\infty$. These boundedness statements can be used with (15) to prove $\tau(t) \in \mathcal{L}_\infty$. And from (7), $\dot{r}(t) \in \mathcal{L}_\infty$ can be ensured, from which $\ddot{q}(t) \in \mathcal{L}_\infty$ can be guaranteed. The remaining terms can be proven as bounded via utilizing standard signal chasing techniques.

Integrating (30) in time from the initial time t_0 to $t = +\infty$ yields

$$\int_{t_0}^{+\infty} \beta \left\| z(t) \right\|^2 dt \leq V(t_0) - V(+\infty) \leq V(t_0) \qquad (31)$$

from which $z(t) \in \mathcal{L}_2$ is proven. Since $z(t), \dot{z}(t) \in \mathcal{L}_\infty$ was shown as well, then from Barbalat's Lemma in [18], [20], $\|z(t)\| \to 0$ as $t \to +\infty$ is guaranteed and thus achieving asymptotic joint position tracking.

V. Simulation Results

Numerical simulations were performed with the dynamic model of a two degree of freedom planar robot manipulator. The dynamic model in (1) was considered with the following terms

$$M = \begin{bmatrix} p_1 + 2p_3c_2 & p_2 + p_3c_2 \\ p_2 + p_3c_2 & p_2 \end{bmatrix} \qquad (32)$$

$$C = \begin{bmatrix} -p_3s_2\dot{q}_2 & -p_3s_2(\dot{q}_1 + \dot{q}_2) \\ p_3s_2\dot{q}_1 & 0 \end{bmatrix} \qquad (33)$$

$$G = \begin{bmatrix} p_4c_1 + p_5c_{12} \\ p_5c_{12} \end{bmatrix} \qquad (34)$$

$$F_d = \begin{bmatrix} p_6 & 0 \\ 0 & p_7 \end{bmatrix} \qquad (35)$$

in which $s_2 = \sin(q_2)$, $c_2 = \cos(q_2)$, $c_{12} = \cos(q_1 + q_2)$, $p_1 = 3.473$, $p_2 = 0.193$, $p_3 = 0.242$, $p_4 = 12.936$, $p_5 = 3.528$, $p_6 = 5.3$, $p_7 = 1.1$. It is highlighted that, when performing the numerical simulations, the above dynamic model was utilized only to simulate the motion of the robot manipulator and was not utilized as part of the control input torque.

The periodic desired joint position vector was selected as

$$q_d = \begin{bmatrix} 0.3\sin(1.5t) \\ 0.5\pi + 0.3\sin(1.5t) \end{bmatrix} \text{rad}. \qquad (36)$$

The robot manipulator is considered to be at rest with the initial joint position as $q(0) = [0.3, 0.3]^T$ rad. The initial values of the entries of the estimated weight matrix were set to zero while hyperbolic tangnt function was chosen as the activation function. The control gains were adjusted via trial and error and the gain of $r(t)$ in (15) is considered to be constant and combined in K_r. Satisfactory tracking performance is obtained when the control gains were chosen as $K_r = 51.5$, $\alpha = 1$, $k_l = 0.1$ and $k_{nn} = 15$.

The results of the simulation are given in Figures 1–4. The joint position tracking error $e(t)$ is given in Figure 1 while the actual and desired joint positions are presented in Figure 2. From Figures 1 and 2 it is clear that the tracking control objective was met. The control input torque is demonstrated in Figure 3 while the entries of the estimated weight matrix are shown in Figure 4.

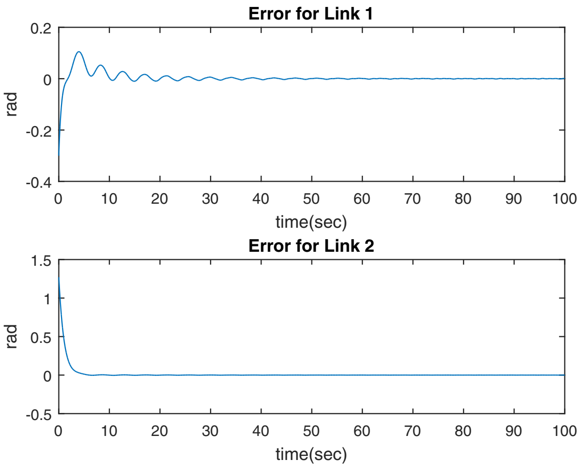

Fig. 1. Joint position tracking error $e(t)$

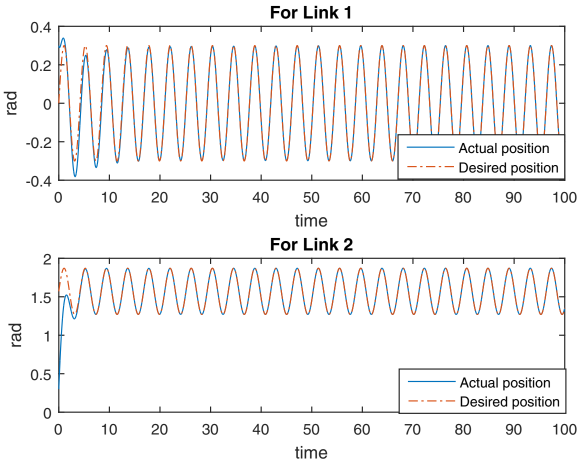

Fig. 2. Joint position $q(t)$ and desired joint position $q_d(t)$

VI. Experimental Results

Preliminary experiments were performed by utilizing the last two links (*i.e.*, links 2 and 3) of the in–house developed 3 degree of freedom planar robot manipulator shown in Figure 5. In the experiments, the desired joint positions given in (36) were utilized. The robot manipulator was considered to be at rest with the initial joint position as $q(0) = [0.0, 0.0]^T$ rad. The entries of the estimated weight matrix were initiated from zero while activation function was chosen as hyperbolic tangent function. Similar to the numerical simulations, the control gains were adjusted via trial and error and the gain of $r(t)$ in (15) is considered to be constant and combined in K_r.

Satisfactory tracking performance was obtained when the control gains were chosen as $K_r = 5$, $\alpha = 2$, $k_l = 0.05$ and $k_{nn} = 15$.

The experiment results are given in Figures 6–9. The joint position tracking error $e(t)$ is given in Figure 6 while the actual and desired joint positions are presented in Figure 7. From Figures 6 and 7 it is clear that the tracking control

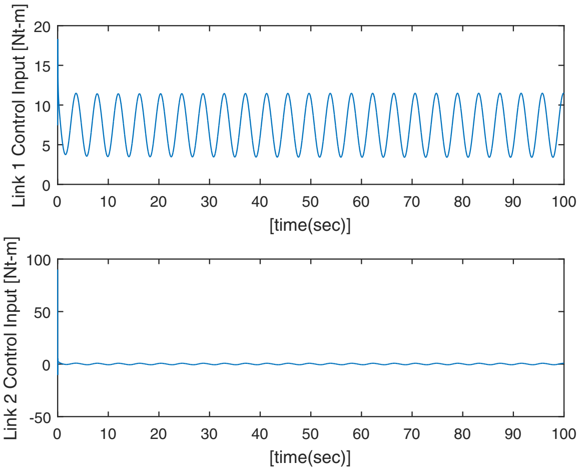

Fig. 3. Control input torque $\tau(t)$

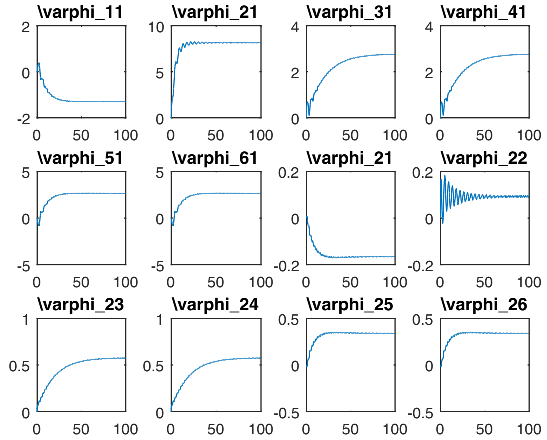

Fig. 4. Entries of the estimated weight matrix $\hat{\varphi}(t)$

objective was met. The control input torque is demonstrated in Figure 8 while the entries of the estimated weight matrix are shown in Figure 9.

VII. Summary, Comparisons and Future Works

In this paper, control of robot manipulators performing a periodic task was considered. Robot manipulator's dynamic model was considered to be unavailable for the control design. Restricted by these, a neural network based repetitive learning controller was designed. Stability of the closed loop system was investigated via Lyapunov type tools and asymptotic stability of the joint position tracking error was guaranteed. Simulations and initial experiments were performed that demonstrated the performance of the proposed controller.

A brief comparison of the proposed study with some of the closest works in the literature will be given. When compared with the saturation function based repetitive learning controller in [9], the proposed controller includes a neural network compensation component which results in the feedback gain of the learning update rule to be reduced significantly. In [9], an adaptive repetitive controller was also designed and when compared with it, the proposed controller is globally model independent and thus does not require a regressor matrix to be obtained. On the other hand, a comparison can be made with some of the neural network controllers in the literature. With the standard neural network controllers, usually only an ultimately bounded result can be obtained mostly because of the functional reconstruction error. An attempt to obtain asymptotic stability with a neural network controller was presented in [21] where asymptotic stability was obtained for a variable structure controller. On the other hand, in our work, asymptotic stability was ensured.

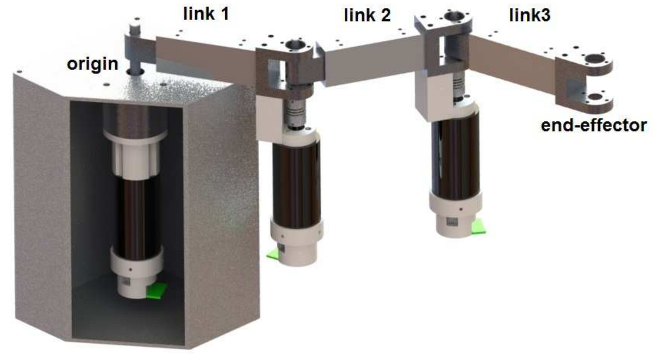

Fig. 5. The 3 degree of freedom robot manipulator

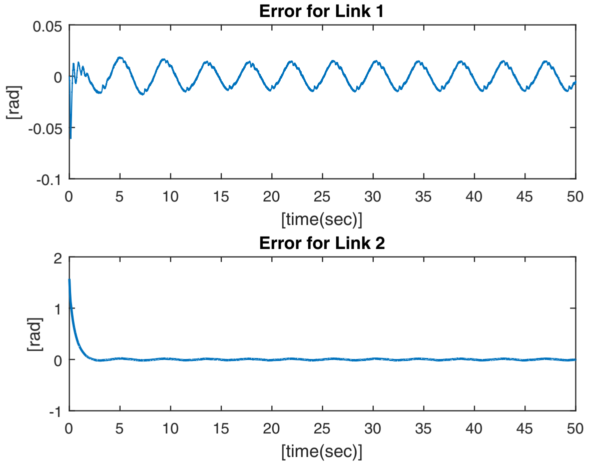

Fig. 6. Joint position tracking error $e(t)$

There are several possible research avenues that may be considered for future work. One line of future research will focus on rewriting the uncertain vector Ω_d with a two layer neural network model. The nonlinearity of the two layer neural networks avoided us to design the update rules for the weight matrices, thus, first future work will be based on modeling the uncertain vector with two layer neural networks. Other line of future work will focus on performing comparative simulations and experiments with some of the closest works in the literature. Designing an output feedback

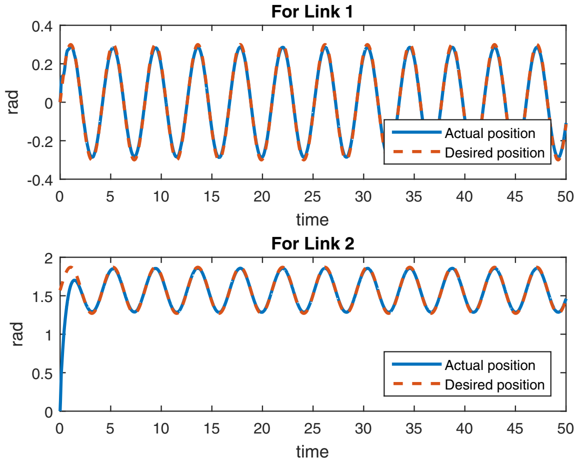

Fig. 7. Joint position $q(t)$ and desired joint position $q_d(t)$

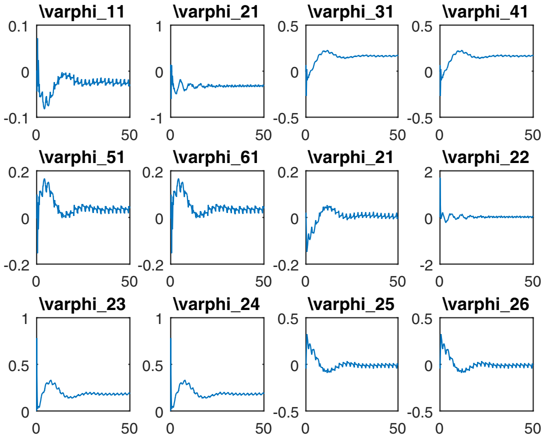

Fig. 9. Entries of the estimated weight matrix $\hat{\varphi}(t)$

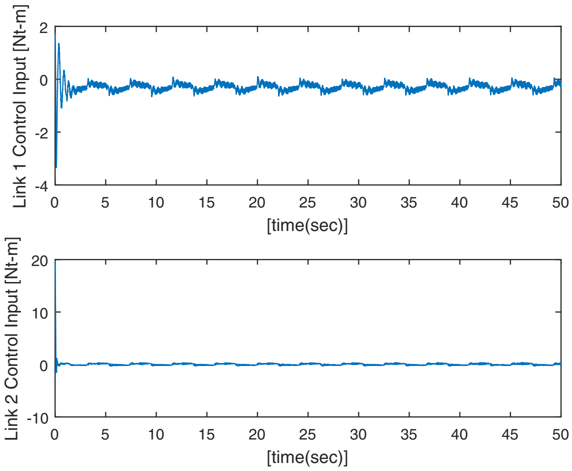

Fig. 8. Control input torque $\tau(t)$

form of the proposed control which is equivalent to removing the need for joint velocity measurements is also aimed.

REFERENCES

[1] G. Hillerstrom and K. Walgama, "Repetitive control theory and applications - a survey," in *Proc. IFAC World Congress*, San Francisco, CA, USA, 1996, pp. 1–6.

[2] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Syst. Mag.*, vol. 26, no. 3, pp. 96–114, 2006.

[3] J.-X. Xu and Y. Tan, *Linear and Nonlinear Iterative Learning Control*. Berlin, Germany: Springer Verlag, 2003.

[4] S. Hara, Y. Yamamoto, T. Omata, and M. Nakano, "Repetitive control system: A new type servo system for periodic exogenous signals," *IEEE Trans. Automat. Contr.*, vol. 33, no. 7, pp. 659–668, 1998.

[5] M.-S. Tsai, M.-T. Lin, and H.-T. Yau, "Development of command-based iterative learning control algorithm with consideration of friction, disturbance, and noise effects,," *IEEE Trans. Contr. Syst. Technol.*, vol. 14, no. 3, pp. 511–518, 2006.

[6] W. Messner, R. Horowitz, W.-W. Kao, and M. Boals, "A new adaptive learning rule," *IEEE Trans. Automat. Contr.*, vol. 36, no. 2, pp. 188–197, 1991.

[7] R. Horowitz, "Learning control of robot manipulators," *ASME J. Dynamic Systems, Measurement, and Control*, vol. 115, pp. 402–411, 1993.

[8] N. Sadegh and R. Horowitz, "Stability and robustness analysis of a class of adaptive controllers for robotic manipulators," *Int. J. Robotics Research*, vol. 9, no. 3, pp. 74–92, 1990.

[9] W. E. Dixon, E. Zergeroglu, D. M. Dawson, and B. Kostic, "Repetitive learning control: A Lyapunov-based approach," *IEEE Trans. Syst., Man, Cybern. B*, vol. 32, no. 4, pp. 538–545, 2002.

[10] P. Tomei and C. M. Verrelli, "Linear repetitive learning controls for nonlinear systems by Pade approximants," *Int. J. Adaptive Control & Signal Processing*, vol. 29, pp. 783–804, 2015.

[11] C. M. Verrelli, "Repetitive learning control design and period uncertainties," *Asian J. Control*, vol. 17, no. 6, pp. 2417–2426, 2015.

[12] S. Scalzi, S. Bifaretti, and C. M. Verrelli, "Repetitive learning control design for led light tracking," *IEEE Tr. on Control Systems Technology*, vol. 23, no. 3, pp. 1139–1146, 2015.

[13] F. Lewis, S. Jagannathan, and A.Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems*. London, UK: Taylor & Francis, 1999.

[14] Y. Kim and F. Lewis, *High-Level Feedback Control with Neural Networks*. Singapore: World Scientific, 1998.

[15] F. L. Lewis, D. M. Dawson, and C. T. Abdallah, *Robot Manipulator Control: Theory and Practice*. New York, NY, USA: Marcel Dekker, Inc., 2004.

[16] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.

[17] F. L. Lewis, "Nonlinear network structures for feedback control," *Asian J. Control*, vol. 1, no. 4, pp. 205–228, 1999.

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

[19] E. Kreyszig, *Advanced Engineering Mathematics*. New York, NY, USA: Wiley, 2011.

[20] H. K. Khalil, *Nonlinear Systems, 3rd Edition*. New York, NY, USA: Prentice Hall, 2002.

[21] Y. H. Kim, F. L. Lewis, and D. M. Dawson, "Intelligent optimal control of robotic manipulators using neural networks," *Automatica*, vol. 36, no. 9, pp. 1355–1364, 2000.