Proceedings of IMECE'01
2001 ASME International Mechanical Engineering Congress and Exposition
November 11-16, 2001, New York, New York, USA

IMECE01/DSC-3B-4

ON THE DESIGN OF ADAPTIVE ROBUST REPETITIVE CONTROLLERS*

Bin Yao Li Xu
School of Mechanical Engineering
Purdue University
West Lafayette, IN 47907
Email: byao@ecn.purdue.edu

ABSTRACT

A new perspective on dealing with the noise sensitive problem of repetitive control algorithms is given in the paper. It is firstly shown that, in continuous-time domain, what the conventional repetitive learning algorithm does is equivalent to adapting all the values of the periodic uncertainties over one period. Such an endeavor means very high bandwidth of the learning algorithm as an infinite number of parameters need to be adapted, which puts a great demand on microprocessor memory in implementing the algorithms. At the same time, such a formulation also makes the algorithm very sensitive to noise as it treats the values of the periodic uncertainties over the same period totally independent from each other, just like a random noise. Based on this new perspective on the noise sensitive problem of repetitive algorithm, a simple remedy is provided for the recently proposed adaptive robust repetitive control (ARRC) design by recognizing the physical dependence of the values of the periodic uncertainties over the same period and using certain known basis functions to capture these physical dependence. By doing so, only the amplitudes of these known basis functions need to be adapted on-line. The net results are that, not only the number of the parameters to be adapted is reduced drastically, but also the noise sensitive problem of the conventional learning algorithm is overcome. The precision motion control of a linear motor drive system is used as an application example. The comparative experimental results demonstrate that, with the new adaptive robust repetitive control design, not only the noise sensitive problem of repetitive learning is completely eliminated, but also a much improved tracking performance is achieved due to the built-in extrapolation capability of the basis functions used.

*THE WORK IS SUPPORTED IN PART BY THE NATIONAL SCIENCE FOUNDATION UNDER THE CAREER GRANT CMS-9734345 AND IN PART BY A GRANT FROM THE PURDUE RESEARCH FOUNDATION.

1 Introduction

When the dynamic model of a nonlinear system is known precisely, many model-based control theories and design methods can be used to develop nonlinear controllers (Krstic et al., 1995) for trajectory tracking. However, due to the model uncertainties, it is difficult to derive the exact description of the system. Recently, there have been many studies in the topic of “repetitive control” for controlling of mechanical systems in an iterative manner during repetitive operations (Hara et al., 1988; Tomizuka et al., 1989; Sadegh et al., 1990; Horowitz, 1993). Repetitive control schemes are easy to implement and do not require exact knowledge of the dynamic model. This control concept arises from the recognition that many tracking systems, such as hard disk drives, rotation machine tools, or robots, have to deal with periodic reference and/or disturbance signals. The basic idea of repetitive control is to improve the tracking performance from one cycle to the next by adjusting the input based on the error signals between the desired motion and the actual motion of the system from the previous cycles. With consecutive iterations, the system is expected to eventually learn the task, and execute the motion without any error. The repetitive control is similar to the iterative learning (Horowitz, 1993; Arimoto et al., 1984; Bondi et al., 1988; Moore et al., 1992) scenario, where the desired trajectory is given in a finite time interval and the same initial setting is required at every learning trial.

Due to the difficulty of stability analysis, repetitive control has been primarily applied to linear systems (Hara et al., 1988; Tomizuka et al., 1989) or linearizable nonlinear systems. The first rigorous stability analysis of a nonlinear repetitive controller was presented in (Sadegh et al., 1990). However, the proposed control law could not handle model uncertainties such as exogenous disturbances which might not be periodic. Realizing this, the authors added a projection mapping in the “learning” algorithm in order to guarantee the boundedness of the repetitive estimate. Although this *ad hoc* method was very important and useful from an implementation point of view, it was not well justified theoretically. Recently, Xu *et al.* proposed a robust learning control scheme (Xu et al., 2000) by combining the design methods of variable structure control with iterative learning control straightforwardly. However, the transient performance of this controller was unknown. The actual system may have large tracking errors during the initial transient period or have a sluggish response. The control law also involved switching in order to achieve asymptotic tracking, which introduces control chattering. Although chattering can be avoided by using some smooth techniques (Slotine and Li, 1991), the convergence of the tracking errors to zero was no longer possible, even when the system is subjected to repeatable uncertainties only. Furthermore, the learning algorithm may still go unbounded under certain circumstances.

In (Xu and Yao, 2001), the idea of adaptive robust control (ARC) (Yao, 1997; Yao and Tomizuka, 1996; Yao and Tomizuka, 1997; Yao and Tomizuka, 2001) was integrated with repetitive control techniques to construct performance oriented control laws for a class of nonlinear systems in the presence of model uncertainties for repetitive operations. Like the robust learning control design in (Xu et al., 2000), we use the repetitive control technique to learn and eliminate the periodic uncertainties as much as possible while employing robust feedback to handle the non-periodic uncertainties. However, our approach is more than just simply putting these two schemes together. Specifically, based on the available bounds on the repeatable uncertainties, the widely used discontinuous projection mapping is utilized to modify the learning algorithm, which guarantees that the repetitive estimates belong to a known bounded region all the time no matter if the system is subject to non-repeatable disturbances or not. As a result, the possible destabilizing effect of the on-line learning in the presence of non-repeatable model uncertainties is avoided, and certain simple robust feedback can be synthesized to attenuate the effect of both repeatable and non-repeatable uncertain nonlinearities effectively for a guaranteed output tracking transient and final tracking accuracy in general. In addition, in the presence of periodic uncertain nonlinearities only, asymptotic output tracking is achieved without using any discontinuous feedback terms.

Generally speaking, repetitive controllers suffer from one major drawback – the repetitive learning algorithm is very sensitive to noise (Tomizuka et al., 1989). For the internal model principle based linear repetitive control algorithms, an effective modification which enhances the stability of repetitive control systems in the presence of noise is to use the so-called Q filters in the repetitive learning algorithm (Tomizuka et al., 1989). In this paper, we approach this problem from an adaptive robust control (ARC) view point (Yao, 1997; Yao and Tomizuka, 1996; Yao and Tomizuka, 1997; Yao and Tomizuka, 2001). It is firstly shown that, in continuous time domain, what the repetitive learning algorithm (Xu and Yao, 2001) does is equivalent to adapting an infinite number of parameters. Such an endeavor not only causes very high bandwidth of the learning algorithm, which may need huge memory in implementation to store the values to be adapted, but also makes the algorithm very sensitive to noise as it treats the values of the periodic uncertainties over the same period totally independent from each other, just like a random noise. Based on this observation, a simple remedy to the problem is to recognize the physical dependence of the values of the periodic uncertainties over the same period and use certain known basis functions to capture these physical dependence. By doing so, only the amplitudes of these known basis functions need to be adapted on-line to learn the actual periodic uncertainties. The net results are that, not only the number of the parameters to be adapted is reduced drastically, which has less demand on the microprocessor memory needed in implementation, but also the noise sensitive problem of the conventional learning algorithm is overcome as the dependence of the values of the known basis function naturally smooths out the effect of random noises on the on-line adaption of the unknown amplitude associated with the basis function.

To verify the effectiveness of the proposed new adaptive robust repetitive control (ARRC) design strategy, the precision motion control of a linear motor drive system is used as a case study. Extensive comparative experimental results demonstrate that, not only the noise problem is indeed eliminated in implementation, but also a much improved tracking performance is obtained due to the extrapolation capability of known basis functions used. In all existing repetitive control algorithms, the tracking error can only be reduced over different periods, but not within the same period. It is experimentally shown that the proposed ARRC is also able to reduce the tracking error over the same period and has a faster convergence of learning.

2 Discontinuous Projection Based Adaptive Robust Repetitive Control (ARRC)

In this section, the recently proposed discontinuous projection based adaptive robust repetitive control (Xu and Yao, 2001) is reviewed in order to introduce the new perspective on the noise sensitive problem of repetitive learning. For simplicity, a first-order nonlinear system in the following form is considered

$$\dot{x} = \varphi(x) + u + d(x,t) \tag{1}$$

where $x \in \mathbb{R}$ is the state of the system, $u \in \mathbb{R}$ is the control input, $\varphi \in \mathbb{R}$ represents an unknown nonlinear function and $d(x,t)$ represents the lumped modelling error and exogenous disturbances. The control objective is to synthesize a bounded control input u such that x tracks a desired trajectory $x_d(t)$ as closely as possible. To achieve this objective, we make the following reasonable and practical assumption:

Assumption 2.1. *The unknown functions $\varphi(x)$ and $d(x,t)$ are bounded by some known functions, i.e.,*

$$\varphi(x) \;\in\; \Omega_\varphi \stackrel{\Delta}{=} \{\, \varphi : \;\; \varphi_{\min}(x) \le \varphi(x) \le \varphi_{\max}(x) \}, \tag{2}$$

$$d(x,t) \;\in\; \Omega_d \stackrel{\Delta}{=} \{\, d : \;\; |d(x,t)| \le d_{\max}(x,t) \}, \tag{3}$$

where $\varphi_{\min}(x)$, $\varphi_{\max}(x)$ and $d_{\max}(x,t)$ are known.

In the above and throughout the paper, the following notations are used: $\bullet_{\min}$ for the minimum value of $\bullet$, $\bullet_{\max}$ for the maximum value of $\bullet$, and the operation $\leq$ for two vectors is performed in terms of the corresponding elements of the vectors. $\hat{\bullet}$ denotes the estimate of $\bullet$.

In practice, many tracking systems have to deal with repetitive tasks, for example, robots are often required to execute the same motion over and over again. Therefore, for these applications, it is assumed that the desired trajectory signal $x_d(t)$ is periodic. Namely,

$$x_d(t-T) = x_d(t), \tag{4}$$

where T is the period. An immediate consequence of equation (4) is that the unknown function $\varphi(x_d(t))$ is also periodic. For simplicity, denote $\varphi(x_d(t))$, $\varphi_{\min}(x_d(t))$ and $\varphi_{\max}(x_d(t))$ as $\varphi_d(t)$, $\varphi_{\min}(t)$ and $\varphi_{\max}(t)$, respectively. Then, we have $\varphi_d(t-T) = \varphi_d(t)$.

Let $\hat{\varphi}_d$ denote the estimate of φ_d and $\tilde{\varphi}_d$ the estimation error (i.e., $\tilde{\varphi}_d \overset{\Delta}{=} \hat{\varphi}_d - \varphi_d$). Under Assumption 2.1, the idea of discontinuous projection based ARC design (Yao, 1997) can be borrowed to solve the tracking control problem for (1). Specifically, with the initial estimate satisfying the physical constraints (2), i.e., $\varphi_{\min}(\tau) \leq \hat{\varphi}_d(\tau) \leq \varphi_{\max}(\tau)$, $\forall \tau \in [-T, 0]$, the following repetitive "learning" algorithm is applied to update the estimate $\hat{\varphi}_d(t)$

$$\hat{\varphi}_d(t) = \mathrm{Proj}_{\hat{\varphi}}(\hat{\varphi}_d(t-T) + \Gamma z), \tag{5}$$

where Γ is the adaptation rate which is a positive scalar in this case, $z = x - x_d$ represents the tracking error, and the projection mapping $\mathrm{Proj}_{\hat{\varphi}}(\cdot)$ is defined by

$$\mathrm{Proj}_{\hat{\varphi}}(\bullet) = \begin{cases} \varphi_{\max}(t) & \text{if} \quad \bullet > \varphi_{\max}(t) \\ \varphi_{\min}(t) & \text{if} \quad \bullet < \varphi_{\min}(t) \\ \bullet & \text{if} \quad \varphi_{\min}(t) \leq \bullet \leq \varphi_{\max}(t) \end{cases} \tag{6}$$

Using similar arguments as in (Yao and Tomizuka, 1996; Sastry and Bodson, 1989), it can be shown that the above type of repetitive learning algorithm has the following properties

$$\textbf{P1} \quad \hat{\varphi}_d \in \Omega_\varphi = \{\hat{\varphi} : \ \varphi_{\min}(t) \leq \hat{\varphi}_d(t) \leq \varphi_{\max}(t)\} \tag{7}$$

$$\textbf{P2} \quad \tilde{\varphi}_d(t)\Big\{\Gamma^{-1}\big[\mathrm{Proj}_{\hat{\varphi}}\big(\hat{\varphi}_d(t-T)+\Gamma z\big)-\hat{\varphi}_d(t-T)\big]-z\Big\} \leq 0 \tag{8}$$

The structure of (1) motivates us to design the adaptive robust repetitive control law as follows:

$$u = u_a + u_s, \quad u_a = -\hat{\varphi}_d(t) + \dot{x}_d, \tag{9}$$

where u_a is an adjustable model compensation need for achieving perfect tracking, and u_s is a robust control law to be specified later. Substituting (9) into (1), and then simplifying the resulting expression, the resulting error dynamics is

$$\dot{z} = -\tilde{\varphi}_d(t) + \Delta\varphi + u_s + d(x,t), \tag{10}$$

where $\Delta\varphi = \varphi(x) - \varphi_d(t)$. From mean-value theorem, it is assumed that the following inequality is satisfied:

$$|\Delta\varphi| = |\varphi(x) - \varphi(x_d)| \leq \rho_\varphi(x,t)|z|, \tag{11}$$

where ρ_φ is a known function. The robust control law u_s consists of two terms given by:

$$u_s = u_{s1} + u_{s2}, \quad u_{s1} = -k_{s1}z, \tag{12}$$

where u_{s1} is used to stabilize the nominal system with k_{s1} being any nonlinear gain satisfying

$$k_{s1} \geq k + \rho_\varphi(x,t), \tag{13}$$

in which k is a positive constant. u_{s2} is a robust feedback used to attenuate the effect of model uncertainties, which is required to satisfy the following two constraints

$$\begin{array}{ll} \text{i} & z[-\tilde{\varphi}_d(t) + d(x,t) + u_{s2}] \leq \varepsilon \\ \text{ii} & z u_{s2} \leq 0 \end{array} \tag{14}$$

where ε is a positive design parameter which can be arbitrarily small.

Theorem 2.1. *If the "learning" algorithm is chosen as (5), then the adaptive robust repetitive control law (9) guarantees that*

A. *In general, all signals are bounded and the tracking error is bounded above by*

$$|z|^2 \leq \exp(-2kt)|z(0)|^2 + \frac{\varepsilon}{k}[1 - \exp(-2kt)], \tag{15}$$

i.e., the tracking error exponentially decays to a ball. The exponential converging rate $2k$ and the size of the final tracking error ($|z(\infty)| \leq \sqrt{\frac{\varepsilon}{k}}$) can be freely adjusted by the controller parameters ε and k in a known form.

B. *If after a finite time t_0, the non-periodic uncertain nonlinearities disappear (i.e., $d(x,t) = 0, \forall t \geq t_0$), then, in addition to result A, zero final output tracking error is also achieved, i.e, $z \longrightarrow 0$ as $t \longrightarrow \infty$.* $\triangle$

Proof: Substituting (12) into (10) yields

$$\dot{z} + k_{s1} z = -\tilde{\varphi}_d(t) + \Delta\varphi + d(x,t) + u_{s2}. \tag{16}$$

Noting (11) and (13), the derivative of a positive semi-definite (p.s.d.) function $V_s = \frac{1}{2}z^2$ is given by

$$\begin{aligned} \dot{V}_s &\leq -k_{s1}z^2 + |\Delta\varphi||z| + z[-\tilde{\varphi}_d(t) + d(x,t) + u_{s2}] \\ &\leq -kz^2 + z[-\tilde{\varphi}_d(t) + d(x,t) + u_{s2}]. \end{aligned} \tag{17}$$

Thus from condition i of (14) we have

$$\dot{V}_s \leq -kz^2 + \varepsilon = -2kV_s + \varepsilon, \tag{18}$$

which leads to (15) and thus proves result A of Theorem 2.1. Now consider the situation in B of Theorem 2.1, i.e., $d(x,t) = 0$, $\forall t \geq t_0$. Choose a positive definite (p.d.) function V_a as

$$V_a = V_s + \frac{1}{2}\Gamma^{-1}\int_{t-T}^{t} \tilde{\varphi}_d^2(\tau)\mathrm{d}\tau. \tag{19}$$

From (17), condition ii of (14) and $\varphi_d(t-T) = \varphi_d(t)$, it follows that

$$\begin{aligned} \dot{V}_a \leq & -kz^2 - z\tilde{\varphi}_d(t) + \Gamma^{-1}[\hat{\varphi}_d(t) - \hat{\varphi}_d(t-T)]\tilde{\varphi}_d(t) \\ & -\tfrac{1}{2}\Gamma^{-1}\left[\hat{\varphi}_d(t) - \hat{\varphi}_d(t-T)\right]^2. \end{aligned} \tag{20}$$

Then, noting (5) and (8), Eqn. (20) becomes

$$\begin{aligned} \dot{V}_a \leq & -kz^2 - z\tilde{\varphi}_d(t) + \Gamma^{-1}\left[\mathrm{Proj}_{\hat{\varphi}}(\hat{\varphi}_d(t-T)+\Gamma z) - \hat{\varphi}_d(t-T)\right]\tilde{\varphi}_d(t) \\ = & -kz^2 + \tilde{\varphi}_d(t)\left\{\Gamma^{-1}\left[\mathrm{Proj}_{\hat{\varphi}}(\hat{\varphi}_d(t-T) + \Gamma z) - \hat{\varphi}_d(t-T)\right] - z\right\} \\ \leq & -kz^2. \end{aligned} \tag{21}$$

This shows that $z \in L_2 \bigcap L_\infty$. It is easy to check that $\dot{z} \in L_\infty$. So, $z(t)$ is uniformly continuous. By Barbalat's lemma, $z \longrightarrow 0$ as $t \longrightarrow \infty$. □

3 Adaptive Robust Repetitive Control (ARC) Revisited

Generally speaking, repetitive controllers suffer from one major drawback – the repetitive learning algorithm is very sensitive to noise. Theoretically, we have shown in the previous section that zero final output tracking error can be achieved in the absence of $d(x,t)$. However, this is just an ideal case which does not exist in practice. Since the signals of every feedback system are contaminated by measurement noise, the best the repetitive controller can do is to reduce the tracking error from cycle to cycle until it reaches the noise level. During the subsequent cycle, the tracking error is actually dominated by noise, which is totally random in a point-to-point sense. But the repetitive learning term still keeps attempting to reduce the tracking error to zero. This attempt to "learn" noise and cancel it can easily drive the system to some undesirable states if not unstable (e.g., severe degree of control input chattering), which has been well documented in the literature and verified in the direct digital implementation of the discontinuous projection based ARRC presented in the previous section (Xu and Yao, 2001). For internal model principle based linear repetitive control algorithms, an effective modification which enhances the stability of repetitive control systems is to use the so-called Q filters in the repetitive learning algorithm (Tomizuka et al., 1989). However, this *ad hoc* method is not well justified theoretically and the convergence of the tracking errors to zero is no longer possible, just like the *ad hoc* modification method we used in implementing the discontinuous projection based ARRC in (Xu and Yao, 2001) where slower sampling rate was intentionally used for repetitive learning part for a less degree of control input chattering.

In the following, we will first show that, from the traditional adaptive robust control (ARC) (Yao, 1997; Yao and Tomizuka, 1996; Yao and Tomizuka, 1997; Yao and Tomizuka, 2001) view point, the adaptive robust repetitive control (ARRC) presented in the previous section is essentially the same as the traditional ARC design (Yao, 1997) with all values of the periodic uncertainties over one period being adapted. The analysis will provide a conceptual link between the various repetitive learning (Hara et al., 1988; Tomizuka et al., 1989; Sadegh et al., 1990; Horowitz, 1993) controls and the traditional adaptive control (Sastry and Bodson, 1989), and will help us better understanding the cause and solution to the noise sensitive of repetitive control designs. For this purpose, rewritten equation (1) as

$$\dot{x} = \varphi_d(t) + u + \Delta\varphi + d. \tag{22}$$

Let the unknown parameter set to be $\theta(\tau), \tau \in [0,T)$ and define $\theta(\tau)$ as $\theta(\tau) = \varphi_d(\tau),\ \forall\tau \in [0,T)$. Since $\varphi_d(t)$ is periodic with a period of T, it can be written as

$$\varphi_d(t) = \int_{0^-}^{T^-} \delta\big(\mathrm{mod}(t-\nu,T)\big)\theta(\nu)\ \mathrm{d}\nu \tag{23}$$

where $\delta(\cdot)$ is the delta function and $\mathrm{mod}(\cdot)$ is the modulus function (i.e., let N be the largest integer such that $NT \leq t-\tau < (N+1)T$. Then $\mathrm{mod}(t-\tau,T) = t-\tau-NT$). Eq. (23) indicates that the unknown periodic function $\varphi_d(t)$ is a linear combination of an infinite number of unknown parameters $\theta(\tau),\ \tau \in [0,T)$ with impulse function $\delta\big(\mathrm{mod}(t-\tau,T)\big)$ as basis functions. Thus it can be dealt with by adapting $\theta(\tau),\ \tau \in [0,T)$ if the previous ARC design techniques (Yao, 1997; Yao and Xu, 1999) are to be used. By doing so, the resulting ARC control law would be (Yao, 1997)

$$u = u_a + u_s, \tag{24}$$

$$u_a = -\int_{0^-}^{T^-} \delta\big(\mathrm{mod}(t-\nu,T)\big)\hat{\theta}(\nu,t)\ \mathrm{d}\nu + \dot{x}_d, \tag{25}$$

where the robust control term u_s is the same as before, $\hat{\theta}(\tau,t)$ represents the estimate of $\theta(\tau)$ at the time t with the parameter adaptation law given by

$$\frac{\partial\hat{\theta}(\tau,t)}{\partial t} = Proj_{\hat{\theta}(\tau,\,t)}\left\{\Gamma\delta\big(\mathrm{mod}(t-\tau,T)\big)z(t)\right\}. \qquad (26)$$

where $Proj_{\hat{\theta}(\tau,\,t)}$ is the parameter projection mapping defined in (Yao, 1997).

To see the equivalence of the parameter adaptation law (26) and the repetitive learning law (5), $\forall\tau\in[0,T)$, integrate (26) as follows (for simplicity, the projection mappings in (26) and (5) are dropped in the following derivations, as the case with these projection mappings can be worked out similarly although the process is very tedious and complicated):

$$\begin{aligned}
\hat{\theta}(\tau,\tau^+) &= \hat{\theta}(\tau,\tau^-) + \textstyle\int_{\tau^-}^{\tau^+}\Gamma\delta\big(\mathrm{mod}(\nu-\tau,T)\big)z(\nu)\,d\nu \\
&= \hat{\theta}(\tau,\tau^-) + \Gamma z(\tau) \\
\hat{\theta}(\tau,t) &= \hat{\theta}(\tau,\tau^+) + \textstyle\int_{\tau^+}^{t}\Gamma\delta\big(\mathrm{mod}(\nu-\tau,T)\big)z(\nu)\,d\nu \\
&= \hat{\theta}(\tau,\tau^+), \qquad\qquad \forall t\in(\tau,\tau+T) \\
&\ \ \vdots \\
\hat{\theta}(\tau,(\tau+kT)^+) &= \hat{\theta}(\tau,(\tau+kT)^-) \\
&\quad + \textstyle\int_{(\tau+kT)^-}^{(\tau+kT)^+}\Gamma\delta\big(\mathrm{mod}(\nu-\tau,T)\big)z(\nu)\,d\nu \\
&= \hat{\theta}(\tau,(\tau+kT)^-) + \Gamma z(\tau+kT) \\
\hat{\theta}(\tau,t) &= \hat{\theta}(\tau,(\tau+kT)^+), \\
&\quad + \textstyle\int_{(\tau+kT)^+}^{t}\Gamma\delta\big(\mathrm{mod}(\nu-\tau,T)\big)z(\nu)\,d\nu \\
&= \hat{\theta}(\tau,(\tau+kT)^+), \quad \forall t\in(\tau+kT,\tau+(k+1)T) \\
&\ \ \vdots
\end{aligned} \qquad (27)$$

Eq. (27) indicates that the estimate of $\theta(\tau)$ or $\varphi_d(\tau)$ remains constant during a single period T and is only updated at the time instance $t=\tau+NT$ by:

$$\begin{aligned}
\hat{\theta}(\tau,(\tau+NT)^+) &= \hat{\theta}\big(\tau,(\tau+NT)^-\big) + \Gamma z(\tau{+}NT) \\
&= \hat{\theta}\big(\tau,(\tau+(N-1)T)^+\big) + \Gamma z(\tau{+}NT)
\end{aligned} \qquad (28)$$

which is the same as the previous repetitive learning algorithm (5) (without projection mapping) given by:

$$\hat{\varphi}_d(t) = \hat{\varphi}_d(t-T) + \Gamma z(t),\ \ \forall t, \qquad (29)$$

when one treats $\hat{\varphi}_d(t)$ as $\hat{\varphi}_d(t)=\hat{\theta}(\tau,(\tau+NT)^+)$ where $\tau=\mathrm{mod}(t,T)$ and $N=\mathrm{floor}(t,T)$ (i.e., $t=\tau{+}NT$ and N is the largest integer such that $NT\le t<(N+1)T$). With these notations, (25) becomes

$$\begin{aligned}
u_a &= -\hat{\theta}\big(\mathrm{mod}(t,T),t\big) + \dot{x}_d \\
&= -\hat{\varphi}_d(t) + \dot{x}_d
\end{aligned} \qquad (30)$$

which is the same as (9).

From the above analysis on the conceptual link between the repetitive learning law (5) and the traditional parameter estimation law (26), the following important observation can be made. Namely, the values of the periodic uncertainty φ_d in one period (i.e., $\varphi_d(\tau),\ \tau\in[0,T)$) are treated totally independent from each other as each value $\theta(\tau)=\varphi_d(\tau)$ is updated by a parameter adaptation law having the form of (26). Such an endeavor not only puts a great demand on the microprocessor memory needed to implement the algorithm, as theoretically an infinite number of parameters need to be updated in each cycle, but also makes the algorithm very sensitive to random noise in implementation, which can be thought due to the vulnerability of the impulse type basis functions (or regressors) in (26) to the random noises. Such an observation also motivates us to look for the following natural addressing strategy. Namely, instead of treating the values of the periodic uncertainty φ_d in one period totally independent from each other, suitable basis functions $\Phi=[\phi_1,\ldots,\phi_n,\ldots]^T$ should be introduced to capture physical dependence of these values and only the unknown amplitudes associated with these basis functions need to be adapted on-line. For example, since $\varphi_d(t)$ is a periodic function with a known period T, it can be represented by a Fourier series

$$\varphi_d(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty}(A_n\cos n\omega t + B_n\sin n\omega t), \qquad (31)$$

where $\omega=2\pi/T$. Consider that the mechanical system is physically equivalent to a low-pass filter with finite bandwidth. $\varphi_d(t)$ is dominated by only finite frequency components. It can be approximated well by a Fourier series with finite terms in practice

$$\varphi_d(t) = \frac{A_0}{2} + \sum_{n=1}^{m}(A_n\cos n\omega t + B_n\sin n\omega t),\quad m<\infty\,. \qquad (32)$$

Thus, with the following finite number of basis functions

$$\Phi = [1,\ \cos\omega t,\ \sin\omega t,\ \cdots,\ \cos m\omega t,\ \sin m\omega t]^T. \qquad (33)$$

the unknown periodic function $\varphi_d(t)$ can be put into a linear combination of these known basis function as

$$\varphi_d(t) = \Phi^T\theta \qquad (34)$$

where $\theta=[A_0/2,\ A_1,\ B_1,\ \cdots,\ A_m,\ B_m]^T$ represent the unknown Fourier coefficients. With such a formulation, only finite number of unknown amplitudes θ need to be adapted on-line and the the smooth basis functions Φ will make the resulting learning algorithm less sensitiveness to random noise.

Remark 3.1. *In practice, we normally have more information about the nature of the unknown periodic function $\varphi_d(t)$. For example, almost all mechanical systems are subject to Coulomb friction which can be described by $A_f\,\mathrm{sgn}(v)$, where v represents the velocity. Thus we know $\varphi_d(t)$ has a component $\mathrm{sgn}(v)$ although its amplitude A_f is unknown. Similarly, for any mechanical systems, the inertia and damping effects may have to be considered for any precision motion control, which have the form of*

$m\dot{v}_d$ and bv respectively, where m and b are the unknown inertia and damping coefficient. In such a case, $\mathrm{sgn}(v)$*, the desired acceleration $\dot{v}_d$, and v can be chosen as some of the basis functions to achieve an effective characterization of $\varphi_d(t)$, i.e., let*

$$\Phi = [\mathrm{sgn}(v),\ \dot{v}_d,\ v,\ 1,\ \cos\omega t,\ \sin\omega t,\ \cdots,\ \cos m\omega t,\ \sin m\omega t]^T. \tag{35}$$

With the linear parametrization (34), equation (1) can be rewritten as

$$\dot{x} = \Phi^T\theta + \Delta\varphi + u + d(x,t), \tag{36}$$

Similar to Assumption 2.1, the following reasonable assumption can be made:

Assumption 3.1. *The unknown parameters θ lie in a known bounded region Ω_θ, i.e.,*

$$\theta \in \Omega_\theta \overset{\Delta}{=} \{\theta : \ \theta_{\min} \le \theta \le \theta_{\max}\}, \tag{37}$$

where $\theta_{\min}$ and $\theta_{\max}$ are known.

Under Assumption 3.1, the discontinuous projection based ARC design (Yao, 1997) can be utilized to solve the tracking control problem for (36) as follows. Similar to (9), the adaptive robust control law is given by:

$$\begin{aligned} u &= u_a + u_s, \\ u_a &= -\Phi^T\hat{\theta} + \dot{x}_d, \\ u_s &= u_{s1} + u_{s2}, \quad u_{s1} = -k_{s1} z, \end{aligned} \tag{38}$$

where $\hat{\theta}$ is the estimate of θ, k_{s1} satisfies (13), and u_{s2} is a robust control law to be specified later. The parameter estimate $\hat{\theta}$ is updated through the following parameter adaptation mechanism

$$\dot{\hat{\theta}} = \mathrm{Proj}_{\hat{\theta}}(\Gamma\Phi z), \tag{39}$$

where $\mathrm{Proj}_{\hat{\theta}}(\bullet) = [\mathrm{Proj}_{\hat{\theta}_1}(\bullet_1), \cdots, \mathrm{Proj}_{\hat{\theta}_m}(\bullet_m)]^T$, and $\bullet_i$ represents the i-th component of $\bullet$. Γ is a positive diagonal matrix in this case. The definition of the projection mapping $\mathrm{Proj}_{\hat{\theta}_i}(\cdot)$ is similar to (6)

$$\mathrm{Proj}_{\hat{\theta}_i}(\bullet_i) = \begin{cases} 0 & \text{if } \hat{\theta}_i = \theta_{i\max} \text{ and } \bullet_i > 0 \\ 0 & \text{if } \hat{\theta}_i = \theta_{i\min} \text{ and } \bullet_i < 0 \\ \bullet_i & \text{otherwise} \end{cases} \tag{40}$$

Substituting (38) into (36), the resulting error dynamics is

$$\dot{z} + k_{s1} z = -\Phi^T\tilde{\theta} + \Delta\varphi + u_{s2} + d(x,t), \tag{41}$$

where $\tilde{\theta} = \hat{\theta} - \theta$ represents the parameter estimation error. The robust feedback u_{s2} is used to attenuate the effect of model uncertainties, which is required to satisfy the following constraints similar to (14)

$$\begin{aligned} &\text{i} \quad z[-\Phi^T\tilde{\theta} + d(x,t) + u_{s2}] \le \varepsilon \\ &\text{ii} \quad z u_{s2} \le 0 \end{aligned} \tag{42}$$

Theorem 3.1. *If the ARC law (38) and parameter adaptation law (39) are applied, the same results as stated in Theorem 2.1 are achieved.*

Proof: The theorem can be proved in the same way as in (Yao, 1997; Yao and Xu, 1999).

4 Experiment Results

To illustrate the above designs, a two-axis precision positioning stage is used as a test-bed (Fig.1). The two axes of the X-Y stage are mounted orthogonally on a horizontal plane with the Y-axis on top of the X-axis. The position of the stage is measured by means of two optical linear encoders with the resolution of 1μm after quadrature. Velocity signal is obtained by averaging the backward differences of the two consecutive position measurements over several samples.

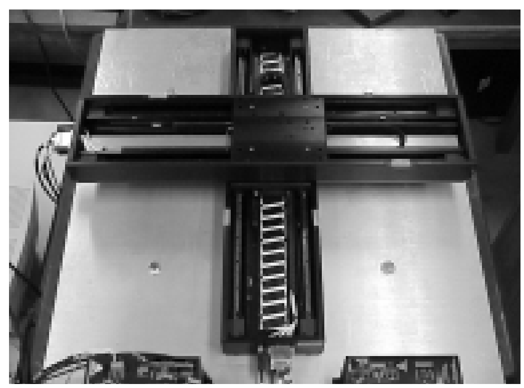

Figure 1. Experimental Setup

Experiment results are obtained for the Y-axis whose simplified dynamics is given by

$$\begin{aligned} \dot{x}_1 &= x_2, \\ M\dot{x}_2 &= u - Bx_2 - F_{fn}(x_2) - F_{EM}(x_1,x_2) + d(x,t), \\ y &= x_1, \end{aligned} \tag{43}$$

where $x = [x_1,\ x_2]^T$ represents the position and velocity of the inertia load, M is the unknown inertia of the payload plus the coil assembly, B is the unknown damping constant, u is the control voltage, F_{fn} is the nonlinear friction force, F_{EM} represents the electro-magnetic force such as cogging force and ripple force, and $d(x,t)$ represents the lumped disturbances consisting of modelling errors and external disturbances.

The controller design proposed in Section 3 can be generalized to solve the tracking control problem of the linear motor. Define a switching-surface-like tracking error metric as

$$\xi = \dot{e}_1 + l_1 e_1 = x_2 - x_{2eq}, \quad x_{2eq} \stackrel{\Delta}{=} \dot{y}_d - l_1 e_1, \tag{44}$$

where y_d is a periodic desired trajectory, $e_1 = y - y_d$, and l_1 is a positive design parameter. Differentiating (44) and substituting the expression given by (43), one obtains

$$M\dot{\xi} = u + \varphi + d(x,t), \tag{45}$$

where $\varphi \stackrel{\Delta}{=} -M\dot{x}_{2eq} - Bx_2 - F_{fn}(x_2) - F_{EM}(x_1,x_2)$. Adding and subtracting the desired compensation term $\varphi_d(t) = -M\ddot{y}_d - B\dot{y}_d - F_{fn}(\dot{y}_d) - F_{EM}(y_d,\dot{y}_d)$ on the right-hand side of (45) yields

$$M\dot{\xi} = u + \varphi_d(t) + \Delta\varphi + d(x,t) \tag{46}$$

where $\Delta\varphi = \varphi - \varphi_d$. As shown in (Sadegh et al., 1990), $\Delta\varphi$ can be bounded by

$$|\Delta\varphi| \leq \gamma_1|e_1| + \gamma_2 e_1^2 + \gamma_3|\xi| + \gamma_4|\xi||e_1|, \tag{47}$$

where $\gamma_1, \gamma_2, \gamma_3$, and γ_4 are positive bounding constants that depend on the desired trajectory and the physical properties of the linear motor.

It is clear that (46) is in a similar form as (22). As such, the adaptive robust repetitive controls proposed in sections 2 and 3 can be applied when repetitive tasks are performed. For example, when the periodic desired compensation term $\varphi_d(t)$ is approximated by (34), (46) becomes

$$M\dot{\xi} = u + \Phi^T\theta + \Delta\varphi + d(x,t), \tag{48}$$

As such, similar to (Yao and Xu, 1999; Xu and Yao, 2000), the adaptive robust control law is given by:

$$\begin{aligned} u &= u_a + u_s \\ &\quad u_a = -\Phi^T\hat{\theta}, \\ &\quad u_s = u_{s1} + u_{s2}, \\ &\qquad u_{s1} = -k\xi - k_e e_1 - k_a e_1^2\xi, \quad u_{s2} = -\frac{1}{4\varepsilon}h^2\xi, \end{aligned} \tag{49}$$

where h is any bounding function satisfying $h \geq \|\theta_{\max} - \theta_{\min}\| \cdot \|\Phi\| + d_{\max}(x,t)$, and the controller parameters k, k_e and k_a are positive scalars satisfying the following conditions:

$$k \geq \frac{1}{2}\gamma_1 + \gamma_3 + \frac{1}{4}\gamma_4, \quad k_e l_1 \geq \frac{1}{2}\gamma_1 + \frac{1}{4}\gamma_2, \quad k_a \geq \gamma_2 + \gamma_4. \tag{50}$$

The associated learning algorithm is $\dot{\hat{\theta}} = \mathrm{Proj}_{\hat{\theta}}(\Gamma\Phi\xi)$. In the experiments, the design parameters of the adaptive robust controller are chosen as:

$$\begin{aligned} &l_1 = 400,\ k = 25,\ k_e = 25,\ k_a = 25,\ \varepsilon = 500,\ d_{\max} = 1, \\ &\theta_{\max} = [2,\ 2,\ 2,\ 0.15]^T,\ \theta_{\min} = -[2,\ 2,\ 2,\ 0]^T, \\ &\hat{\theta}(0) = [0,\ 0,\ 0,\ 0.1]^T,\ \Gamma = 1000 \cdot \mathbf{diag}[1,\ 1,\ 1,\ 0], \\ &\Phi = [1, \cos\omega t, \sin\omega t, \mathrm{sgn}(\dot{x}_2)]^T. \end{aligned} \tag{51}$$

This controller is referred to as the adaptive robust controller (ARC) in the following.

Similarly, the adaptive robust repetitive control scheme proposed in Section 2 can be applied. The control law has the same form as (49) except to replace u_a by $u_a = -\hat{\varphi}_d(t)$ and update $\hat{\varphi}_d(t)$ by $\hat{\varphi}_d(t) = \mathrm{Proj}_{\varphi}(\hat{\varphi}_d(t-T) + \Gamma\xi)$. In the experiments, the design parameters are chosen as:

$$\begin{aligned} &l_1 = 200,\ k = 20,\ k_e = 20,\ k_a = 20,\ \varepsilon = 1,\ \Gamma = 50, \\ &\varphi_{\max} = 2,\ \varphi_{\min} = -2,\ d_{\max} = 1,\ \hat{\varphi}_d(0) = 0, \end{aligned} \tag{52}$$

This controller is referred to as the adaptive robust repetitive controller (ARRC) in the following.

For comparison purpose, a discrete-time repetitive controller (Tomizuka et al., 1989; Tsao and Tomizuka, 1994; Tomizuka, 1993) which was proposed to solve a set-point regulation problem is also implemented:

$$G_r(q^{-1}) = \frac{q^{-N}(1 - 1.96q^{-1} + 0.9608q^{-2})Q(q^{-1})}{(0.0151q^{-1} - 0.01431q^{-2})(1 - Q(q^{-1})q^{-N})}, \tag{53}$$

where q is the one step delay shift operator, $N = 100$, and the Q filter is chosen as $Q(q^{-1}) = (q + 2 + q^{-1})/4$. This controller is referred to as the discrete time repetitive controller (DTRC) in the following.

All the three controllers are implemented using a dSPACE DS1103 controller board. The controllers execute programs at a sampling rate of $f_s = 10$kHz. For ARRC, to reduce the noise effect, as discussed at the beginning of section 3, a slower sampling rate of 2.5kHz is used in discretizing repetitive learning law in implementation.

First we consider a set-point regulation problem when the linear motor is disturbed by a sinusoidal disturbance $d(t) = 0.2\sin(\omega t)(\mathrm{V})$ with $\omega = 200\pi$ rad/s. The performance of the three controllers is shown in Fig.2, Fig.3 and Fig.4, respectively. It can be seen that the controllers can eliminate the effect of disturbance after several periods. However, the proposed ARC type repetitive controllers have a better transient performance. The ARC controller with smooth basis functions has a smaller transient error and the error converges faster.

The motor is then controlled by the two proposed ARC type repetitive controllers to track a desired trajectory $y_d = 0.05[1 - \cos(\omega t)]$ with $\omega = 2\pi$ rad/s; the discrete time repetitive controller did not work well for this case as it was designed for set-point regulation only. The tracking errors are shown in Fig.5. It can be seen that both repetitive ARC controllers achieve a good tracking performance, but the tracking performance of the repetitive ARC with smooth basis functions has a much better tracking performance, especially during the transient period. The tracking error is less than 20μm during the transient period and converges quickly (even much less than one period of $T = 1sec$) to the final tracking error of only $6 \sim 7\mu$m. Since the ARRC uses a point-to-point learning algorithm, it normally takes several periods to for the tracking error to converge. As seen from Fig.5, the tracking error of ARRC converges in two periods (T=1 sec). Comparatively, the tracking error of ARC with smooth basis functions converges in a very short time, which is due to the extrapolation capability of the smooth basis functions, as apposed to the non-extrapolation of the delta basis functions of point-to-point ARRC. At the same time, the smooth basis functions has a better ability in evening out the effect of random noise on the parameter estimates. As such, the smooth basis function based ARC repetitive controller also achieves a better final tracking accuracy as seen from Fig.5.

To test the performance robustness of the algorithms to parameter variations, the motor is run with a 9.1kg payload mounted on it. The tracking errors are given in Fig.6. It shows that almost the same tracking performance as in no-load situation is achieved in spite of the change of inertia load.

Finally, to test the performance robustness of the controllers to disturbance, a step disturbance (a simulated 0.5V electrical signal) is added at about t = 4s. The transient tracking errors become large (more than 50μm) when the disturbance occurs (Fig.7). But the learning algorithm and the parameter adaptation law (Fig.8) are able to capture its effect quickly and generate corresponding control efforts to compensate for it. As such, the tracking error converges quickly after a short transient period.

5 Conclusion

In this paper, the noise sensitive problem of our recently proposed adaptive robust repetitive control (ARRC) law has been carefully re-examined from the traditional parameter adaptation view point. It is shown that, in continuous-time domain, what the conventional repetitive learning algorithm does is equivalent to adapting all the values of the periodic uncertainties over one period with impulse type functions as basis functions. Such a formulation makes the algorithm very sensitive to noise due to the vulnerability of impulse type basis functions to random noises. A simple remedy to this noise sensitive problem is to use smooth basis functions with unknown amplitudes to approximate the unknown periodic terms. Such a formulation also helps to capture the physical dependence of the values of the unknown periodic term over one period and provides certain extrapolation capability that enables the tracking error to converge even during a single period. The precision motion control of a linear motor drive system is used as an application example. The comparative experimental results demonstrate that, with the new adaptive robust repetitive control design, not only the noise sensitive problem of repetitive learning can be easily taken care of, but also a much improved tracking performance is achieved.

REFERENCES

Arimoto, S., Kawamura, T., and Miyazaki, F. (1984). Bettering operation of robots by learing. *Journal of Robotic Systems*, 1(2):123–140.

Bondi, P., Casalino, G., and Gambardella, L. (1988). On the iterative learning control theory for robotic manipulators. *IEEE Journal of Robotics and Automation*, 4:14–22.

Hara, S., Yamamoto, Y., Omata, T., and Nakano, M. (1988). Repetitive control system: a new type servo system for periodic exogenous signals. *IEEE Trans. on Automatica Control*, 33:659–668.

Horowitz, R. (1993). Learning control of robot manipulators. *ASME J. of Dynamic Systems, Measurement, and Control*, 115:402–411.

Krstic, M., Kanellakopoulos, I., and Kokotovic, P. V. (1995). *Nonlinear and adaptive control design*. Wiley, New York.

Moore, K. L., Dahleh, M., and Bhattacharyya, S. P. (1992). Iterative learning control: a survey and new results. *Journal of Robotic Systems*, 9:563–594.

Sadegh, N., Horowitz, R., Kao, W. W., and Tomizuka, M. (1990). A unified approach to the design of adaptive and repetitive controllers for robotic manipulators. *ASME J. of Dynamic Systems, Measurement, and Control*, 112(4):618–629.

Sastry, S. and Bodson, M. (1989). *Adaptive Control: Stability, Convergence and Robustness*. Prentice Hall, Inc., Englewood Cliffs, NJ 07632, USA.

Slotine, J. J. E. and Li, W. (1991). *Applied nonlinear control*. Prentice Hall, Englewood Cliffs, New Jersey.

Tomizuka, M. (1993). On the design of digital tracking controllers. *ASME J. Dynamic Systems, Measurement and Control (50th Anniversary Issue)*, 115(2(B)):412–418.

Tomizuka, M., Tsao, T. C., and Chew, K. K. (1989). Analysis and synthesis of discrete time repetitive controllers. *ASME J. of Dynamic Systems, Measurement and Control*, 111:353–358.

Tsao, T. C. and Tomizuka, M. (1994). Robust adaptive and repetitive digital control and application to hydraulic servo for noncircular machining. *ASME J. Dynamic Systems, Measurement, and Control*, 116:24–32.

Xu, J. X., Viswanathan, B., and Qu, Z. (2000). Robust learning control for robotic manipulators with an extension to a class of nonlinear systems. *International Journal of Control*, 73(10):858–870.

Xu, L. and Yao, B. (2000). Coordinated adaptive robust contour tracking of linear-motor-driven tables in task space. In *Proc. of IEEE Conf. on Decision and Control*, pages 2430–2435, Sydney.

Xu, L. and Yao, B. (2001). Adaptive robust repetitive control of a class of nonlinear systems in normal form with applications to motion control of linear motors. In *Proc. of IEEE/ASME International Conference on Advanced Intelligent Mechantronics (AIM)*, pages 527–532, Como, Italy.

Yao, B. (1997). High performance adaptive robust control of nonlinear systems: a general framework and new schemes. In *Proc. of IEEE Conference on Decision and Control*, pages 2489–2494, San Diego.

Yao, B. and Tomizuka, M. (1996). Smooth robust adaptive sliding mode control of robot manipulators with guaranteed transient performance. *Trans. of ASME, Journal of Dynamic Systems, Measurement and Control*, 118(4):764–775. Part of the paper also appeared in the Proc. of 1994 American Control Conference.

Yao, B. and Tomizuka, M. (1997). Adaptive robust control of

SISO nonlinear systems in a semi-strict feedback form. *Automatica*, 33(5):893–900. (Part of the paper appeared in Proc. of 1995 American Control Conference, pp2500-2505, Seattle).

Yao, B. and Tomizuka, M. (2001). Adaptive robust control of MIMO nonlinear systems in semi-strict feedback forms. *Automatica*, 37(9):1305–1321. Parts of the paper were presented in the *IEEE Conf. on Decision and Control*, pp2346-2351, 1995, and the *IFAC World Congress*, Vol. F, pp335-340, 1996.

Yao, B. and Xu, L. (1999). Adaptive robust control of linear motors for precision manufacturing. In *The 14th IFAC World Congress, Vol. A*, pages 25–30, Beijing. The revised full version will appear in the *International Journal of Mechatronics*.

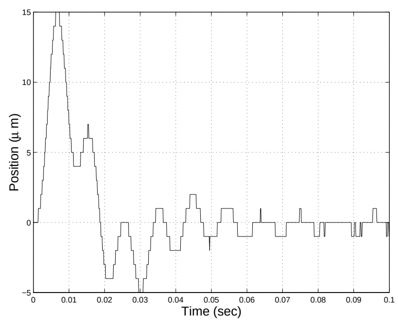

Figure 2. Set-point regulation: discrete-time repetitive controller

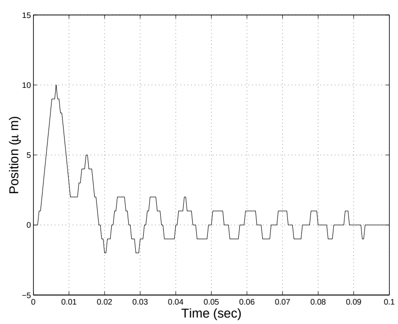

Figure 3. Set-point regulation: adaptive robust repetitive controller

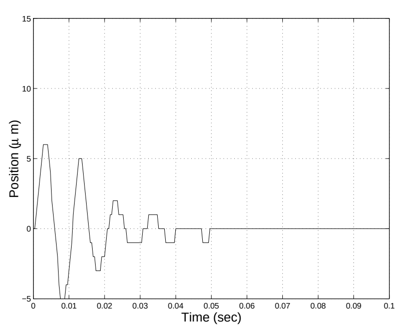

Figure 4. Set-point regulation: adaptive robust controller

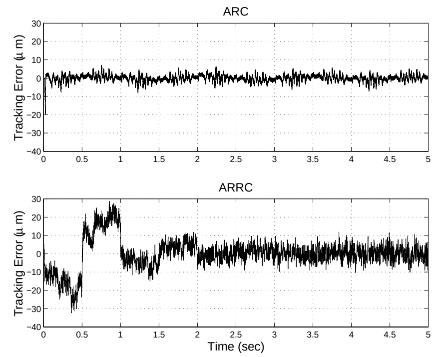

Figure 5. Controller tracking errors (without load)

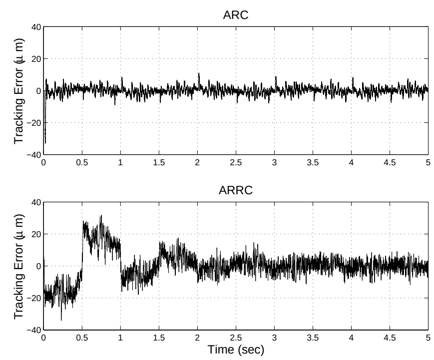

Figure 6. Controller tracking errors (with load)

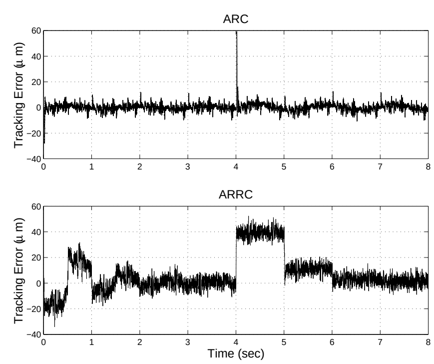

Figure 7. Controller tracking errors (with a step disturbance)

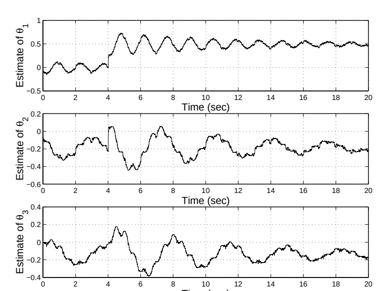

Figure 8. Parameter estimation (with a step disturbance)