Repetitive and Adaptive Control of Robot Manipulators with Velocity Estimation

Kazuma Kaneko and Roberto Horowitz, Member, IEEE

Abstract—This paper presents repetitive and adaptive motion control schemes for rigid-link robot manipulators, when the manipulator’s joint velocities cannot be measured by the control system. The control objective consists in tracking a prescribed desired trajectory. In the case of repetitive control, the desired trajectory is periodic and it is required that the robot achieve the control objective through repeated learning trials. We assume that the robot inverse dynamics are totally unknown, except that they can be represented by an integral of the product of known differentiable kernel function and an unknown influence function. In the case of adaptive control, it is assumed that only the manipulator inertia parameters are unknown and that the desired trajectory jerks are available to the control system. In both control schemes, a velocity observer, which is formulated based on the desired input/output relation of the manipulator, is used to estimate the manipulator joint velocities. A stability analysis of the repetitive and adaptive control schemes with velocity estimation is presented. Simulation and experimental results show that the proposed repetitive control algorithm is successful in achieving the control objective without direct measurement of the joint velocities.

Index Terms—Adaptive control, adaptive observers, learning control systems, manipulators, robots.

I. INTRODUCTION

This paper presents repetitive and adaptive motion control schemes for rigid link robot manipulators connected by rotary and/or spherical joints, when no direct measurement of the manipulator’s joint velocity vector is available to the controller. The control objective consists in tracking a prescribed desired trajectory.

Most adaptive and learning control schemes for robot manipulators require the measurement of both the joint position and velocity vectors to guarantee the asymptotic convergence of the control algorithm. In fact, in most rigorously proven adaptive schemes, the joint velocity signal is used to stabilize the robot closed-loop dynamics and the adaptation signal is a linear combination of the joint position and velocity error vectors. Thus, the simultaneous estimation of the joint velocity vector and robot inverse dynamics has remained a problem of interest to researchers in the robot control community.

The problem of designing nonadaptive controllers for robot manipulators with state observers has been considered in [1]–[6] and references therein. In [2], a smooth nonlinear observer is considered, while in [1] a sliding observer is utilized. In [3], a nonlinear observer based on the robot dynamics is used, while [4] considered the use of a simple linear observer with high gain output injection. Berghuis [6] presents a very comprehensive review of controllers for robot arms with state observation and considers both passivity-based feedback linearization-based motion controllers with state observation. In all these works, the asymptotic convergence to zero of the tracking error norms is assured only if the parameters of the manipulators are exactly known.

Adaptive tracking controllers for robot manipulators with state observers have been considered by [7] and [6] and their bibliographies. Reference [7] proposed an interesting scheme which combines a passivity based adaptive controller with a sliding observer under robust deterministic nonlinear control and showed the local asymptotic convergence of the tracking errors and velocity estimation errors. The scheme presented in [7] requires the on-line computation of the manipulator inverse dynamics function, which can be very computational intensive. Moreover, the asymptotic stability results in [7] are not preserved if the switching functions in the robust deterministic nonlinear control law are replaced by saturation functions. An interesting scheme for the design of adaptive tracking controllers and velocity estimators for magnetic levitated systems is presented in [8]. Unfortunately, the stability of the scheme in [8] can only be rigorously proven if the inertia matrix of the system is constant. As discussed in [7] and [6], there does not appear to be to this date any rigorous asymptotic or exponential convergence result for a smooth-observer-based adaptive or learning controller for robot manipulators. In this paper, we present a rigorous stability analysis for adaptive and repetitive learning controllers for robot manipulators which have a smooth velocity observer and control law.

The adaptive scheme introduced in this paper is a modification of the desired compensation adaptive law (DCAL) introduced in [9], while the repetitive control scheme is similar to the one introduced in [10], with the important difference that the controllers introduced in this paper do not use the joint’s velocity signals. The repetitive control scheme in this paper was first presented in [11] and [12]. A simple linear observer, based on the desired input/output relation of the manipulator, is used to estimate the velocity vector. This simple observer has also been used in [6]. An important and perhaps restrictive assumption used in this paper is that the desired trajectory accelerations are differentiable and their derivatives (the desired trajectory jerks) are accessible to the
control system. This assumption is not unrealistic when the desired trajectories are known in advance, which is the case of many industrial applications.

This paper is organized as follows: Section II formulates the tracking control problem considered in this paper. The DCAL scheme in [9] and the learning repetitive control scheme in [10] are also briefly discussed in this section. In Section III, a smooth observer is presented to estimate the velocity signals, and new adaptive and repetitive controllers are proposed. The stability of these schemes is proven in Section IV. An observer-based version of the delayed repetitive learning algorithm originally introduced in [10] is considered in Section V. This algorithm is particularly useful in real-time digital implementations. Its stability is also rigorously proven. Simulation and experimental results using the Berkeley/NSK two-link SCARA robot arm are presented in Section VI. Conclusions are given in Section VII.

II. ROBOT-MANIPULATOR TRACKING CONTROL

In this paper, we consider robot manipulators with rigid links connected through rotary or spherical joints. Furthermore, it is assumed that each degree of freedom of the manipulator is powered by an independent torque source. Using the Lagrangian formation, the equations of motion for a $n$ degree-of-freedom manipulator may be expressed by

\[
\begin{align*}
M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) &= \tau(t) \\
\end{align*}
\]

where $q, \dot{q}$, and $\ddot{q}$ are the $n \times 1$ vector of joint positions, velocities, and accelerations, respectively. $M(q)$ is an $n \times n$ symmetric, bounded, positive definite matrix function, which is also called generalized inertia matrix. $C(q, \dot{q})\dot{q}$ is the $n \times 1$ vector resulting from Coriolis and centripetal accelerations, $g(q)$ is the $n \times 1$ vector of generalized gravitational forces, and $\tau(t)$ is the $n \times 1$ vector of torque and forces supplied by the actuators. We assume that the matrix $M(q)$ has been defined such that the matrix $\dot{M}(q) - 2C(q, \dot{q})$ is skew-symmetric [9].

In order to derive adaptive and learning tracking control laws, it is convenient to define the inverse dynamic function $w_{\tau}(\cdot): R^{6n} \rightarrow R^6, w_{\tau}(\cdot) \in C^{\infty}$, as follows. For any set of $n \times 1$ vectors $v_1, v_2, v_3, v_4$:

\[
\begin{align*}
\tau(t) &= w_{\tau}(u(t)) = F_p \dot{q} + F_v e_v + \sigma e_r e_v \\
\end{align*}
\]

where $u(t): [\dot{q}(t), \ddot{q}(t), \dddot{q}(t)]^T$.

To simplify the analysis that follows, it is convenient to define the reference velocity signal $\dot{q}_r$ and the reference velocity error signal $e_v = \dot{q} - \dot{q}_r$. [9], [13], respectively, by

\[
\begin{align*}
\dot{q}_r &= \dot{q}_d - \lambda \dot{q}_d \\
e_v &= \dot{q} - \dot{q}_r, \lambda > 0
\end{align*}
\]

Using (5), the tracking error dynamics of the system presented by (1) is given by

\[
\begin{align*}
\dot{e}_v &= e_v - \lambda \dot{e}_v \\
M(q)\dot{e}_v + C(q, \dot{q})e_v &= \tau - w_{\tau}(q, \dot{q}, \ddot{q}, \dddot{q})
\end{align*}
\]

where

\[
\dddot{q}_r = \dddot{q}_d - \lambda \dddot{q}_d
\]

is the reference acceleration vector.

To make our notation more compact, it is convenient to define the extended exogenous desired trajectory vector:

\[
u(t) := [\dddot{q}(t), \dddot{q}^T(t), \dddot{q}_r(t)]^T.
\]

In order to make the derivation of our new learning algorithm easier to follow, we first briefly review the DCAL introduced in [9] and the repetitive learning law introduced in [10]. Both control laws assume that the velocity signal $\dot{q}$ is measurable and are based in the following control law:

\[
\tau = w_{\tau}(u) - F_p \dot{q} - F_v e_v - \sigma e_r e_v
\]

where $w_{\tau}(\cdot): R^{6n} \rightarrow R^6$ is the estimate of the manipulator’s inverse dynamics function $w_{\tau}(\cdot)$ in (3). $F_p$ and $F_v$ are positive definite gain matrices, and $\sigma > 0$. The first term in (9) is a purely feedforward linearization term, while the last three terms are nonlinear position error and velocity error feedback terms.

We now discuss two methods for estimating the inverse dynamics function $w_{\tau}(\cdot)$ in (3). The first is a parametric adaptive control technique, while the second is a repetitive learning technique.

A. Parametric Adaptive Control

This approach is based on the following assumption.

\textbf{Assumption 1}: The inverse dynamics function $w_{\tau}(\cdot)$ can be expressed as

\[
\begin{align*}
w_{\tau}(\cdot) &= W_{\tau}(\cdot)\Theta \\
\end{align*}
\]

where the matrix function $w_{\tau}(\cdot): R^{6n} \rightarrow R^{6n}$ is known and the constant parameter vector $\Theta \in R^{6n}$ is unknown.

Assumption 1 is commonly referred to as the linear parametrization assumption and is frequently used in most robotic adaptive control works.

The following adaptation algorithm is used in the DCAL [9] to generate the inverse function estimate $\hat{w}_{\tau}(\cdot)$:

\[
\hat{w}_{\tau}(\cdot) = W_{\tau}(\cdot)\dot{\Theta}, \quad \dot{\Theta} = \Gamma_{\theta} W_{\tau}^T(u)e_v
\]

where $\Gamma_{\theta}$ is a positive definite gain.
B. Repetitive Learning Control

If the robot is required to track a single periodic trajectory with known period \( T \), it is possible to estimate the function \( \mathbf{w}_r(t) \) directly without explicitly knowing the matrix function \( \mathbf{W}(t) \). In this case, the exogenous desired trajectory vector \( \mathbf{u}(t) \) defined in (8) and the inverse dynamics function \( \mathbf{w}(\cdot) \) defined in (2) can be considered to be periodic functions. Thus, we can consider the inverse dynamics function as an explicit function of time and define the unknown periodic function \( \mathbf{w}_r(\cdot): \mathbb{R}^+ \rightarrow \mathbb{R}^n \) by

\[
\mathbf{w}_r(t) = \mathbf{w}_r^0(t), \quad \mathbf{w}_r(t + T) = \mathbf{w}_r(t), \tag{12}
\]

The repetitive learning law introduced in [10] is based on the following assumption.

Assumption 2: The function \( \mathbf{w}_r(\cdot) \) can be represented by the following linear integral equation of the first kind:

\[
\mathbf{w}_r(t) = \int_0^T K(t, \tau) \mathbf{c}(\tau) d\tau \tag{13}
\]

where \( K(\cdot, \cdot): [0, T] \times [0, T] \rightarrow \mathbb{R} \) is a known nondegenerate kernel which satisfies \( K(t + T, \tau) = K(t, \tau) \), and

\[
k_L = \int_0^T K(t, \tau)^2 d\tau < \infty \tag{14}
\]

and \( \mathbf{c}(\cdot): [0, T] \rightarrow \mathbb{R}^n \) is the unknown influence function.

In (13), we are assuming that both \( \mathbf{w}_r(\cdot) \) and \( \mathbf{c}(\cdot) \) are unknown functions and that a kernel function, \( K(\cdot, \cdot) \), can be selected rather arbitrarily such that (13) is satisfied. The following Lemmas provide conditions under which Assumption 2 is satisfied.

Lemma 1 [10]: Consider a kernel satisfying the Dirichlet conditions defined by

\[
K(t, \tau) = c_0 + \sum_{n=1}^{\infty} \left[ c_n \cos(n\tau) \right] \cos(n\tau) + d_n \sin(n\tau) \sin(n\tau) \tag{15}
\]

where \( n^2 \) is an eigenvalue of the kernel \( K(\cdot, \cdot) \). The kernel \( K(\cdot, \cdot) \) is periodic if \( K(t + T, \tau) = K(t, \tau) \), and

\[
k_L = \int_0^T K(t, \tau)^2 d\tau < \infty \tag{14}
\]

for all \( t \). The repetitive learning law introduced in [10] is given by

\[
\mathbf{\tau}(t) = \mathbf{\dot{w}}_r(t) - \mathbf{F}_r(\mathbf{c}_r(t)) - \mathbf{F}_r \mathbf{e}_r(t) - \sigma |\mathbf{c}_r(t)|^2 \mathbf{c}_r(t) \tag{16}
\]

where the repetitive inverse dynamics function estimate \( \mathbf{\dot{w}}_r(\cdot): \mathbb{R}^+ \rightarrow \mathbb{R}^n \) is given by

\[
\mathbf{\dot{w}}_r(t) = \int_0^T K(t, \tau) \mathbf{c}(\tau) d\tau \tag{17}
\]

and \( \mathbf{F}_r \) is a positive definite gain. Notice that in the repetitive learning law in (17) and (18), \( K(t, \tau) \) plays the role of a functional regressor, while the influence function estimate \( \mathbf{c}(\cdot) \) plays the role of the unknown parameter estimate.

In most implementations of the repetitive control algorithm, we will use kernels which have a finite eigenvalue expansion.

Assumption 3: \( K(\cdot, \cdot) \) in (13) has a finite eigenfunction expansion:

\[
K(t, \tau) = \sum_{n=1}^{N} \lambda_n \psi_n(t) \psi_n(\tau) \tag{19}
\]

where \( \lambda_n \geq 0 \) for \( n = 0, 1, 2, \ldots, N \) and the \( \psi_i \)'s are orthonormal over \([0, T]\). If the kernel \( K(t, \tau) \) satisfies Assumption 3, then Assumption 2 is also satisfied. Assumption 3 also allows us to postulate that the kernel \( K(t, \tau) \) is a persistently exciting (PE) kernel [15]. The kernel \( K(t, \tau) \) is PE since, for all influence functions \( \mathbf{c}(\cdot): \mathbb{R} \rightarrow \mathbb{R}^n \) with a finite eigenfunction expansion and \( \mathbf{\dot{w}}_r(t) = \int_0^T K(t, \tau) \mathbf{c}(\tau) d\tau \), there exist \( 0 < \alpha_1, \alpha_2 \) such that

\[
\alpha_2 \int_0^T |\mathbf{c}(\tau)|^2 d\tau \geq \int_t^{t+T} |\mathbf{\dot{w}}_r(\tau)|^2 d\tau \geq \alpha_1 \int_0^T |\mathbf{c}(\tau)|^2 d\tau \tag{20}
\]

for all \( t \).

Remark: This assumption is not very restrictive in practice since the repetitive signal \( \mathbf{\dot{w}}_r(t) \) can be decomposed into two components \( \mathbf{\dot{w}}_{rF}(t) + \mathbf{\dot{w}}_{rD}(t) \), where \( \mathbf{\dot{w}}_{rF}(t) \) has a finite eigenfunction expansion and satisfies (13), with \( K(t, \tau) \) satisfying (19) and \( \mathbf{c}(\cdot) \) being bounded. The term \( \mathbf{\dot{w}}_{rD}(t) \) contains high frequency components and can be considered as a disturbance input to the control system. In actual implementations, where the estimate functions \( \mathbf{\dot{w}}_r(\cdot), \mathbf{\dot{c}}(\cdot) \), and the kernel \( K(\cdot, \cdot) \) are discretized into finite elements, Assumption 3 is always satisfied. Thus, it is only necessary to determine a sufficiently high degree of discretization so that the term \( \mathbf{\dot{w}}_{rD}(t) \) is small enough. It should be emphasized that the knowledge of the eigenfunction expansion in (19) is not necessary. Assumption 3 is needed so that we can postulate the existence of \( \lambda_{\min} > 0 \) in the kernel eigenfunction expansion in (19). If \( \mathbf{\dot{w}}_r(t) \) is infinite dimensional, then \( \lim_{n \to \infty} \lambda_n = 0 \).
III. ADAPTIVE AND REPETITIVE LEARNING CONTROL WITH STATE OBSERVATION

To implement the parametric adaptive control algorithm and the repetitive control algorithm discussed in the previous section, it is necessary that the joint velocity error vector \( \dot{e}_p \) be measurable. In these algorithms, the reference velocity error signal \( e_r \) given by (5) is used in the feedback terms of the control laws of (9) and (16), to stabilize the manipulator dynamics, and is also used as the adaptation error signal in both the parametric adaptive law (11) and the repetitive learning law (18). In this section, we assume that the position tracking error vector \( e_p \) is directly measurable by the control system, but the velocity error signal \( \dot{e}_p \) is not. We will introduce new parametric adaptive and repetitive learning control laws which do not require a direct measurement of the manipulator joint velocities.

In order to estimate the manipulator joint velocity vector, we introduce the following observer:

\[
\begin{align*}
\dot{q}_p &= q - \hat{q}_p, \\
\dot{\dot{q}}_p &= \dot{q} - \dot{\dot{q}}_p
\end{align*}
\]

where \( \hat{q}_p \) is the estimate of the joint positions and \( \dot{\dot{q}}_p \) is the estimate of the joint velocities. \( \Lambda_\nu \) is a positive definite gain matrix, and \( \lambda_\nu \) is the positive scalar constant gain in (5). This observer structure has also been used in [6].

\begin{align*}
\text{are the joint position and joint velocity estimation errors, respectively.}
\end{align*}

Utilizing the joint position and velocity estimates, we now define the reference velocity error estimate as

\[
\hat{e}_r = \dot{\hat{q}}_p - \dot{q}_d + \lambda_\nu(\hat{q}_p - q_d) = \dot{e}_r + \lambda_\nu e_r
\]

and the auxiliary error signal

\[
z_p = e_p + \dot{\hat{q}}_p
\]

which is the sum of the position tracking and estimation error signals.

A. Parametric Adaptive Control

In order to implement the DCAL adaptive law without requiring measurement of the velocity signals, it is necessary to introduce the following assumption regarding the exogenous desired vector \( \dot{u}(t) \) in (8).

**Assumption 4:** \( \dot{u}, \ddot{u} \in L_{\infty} \), and \( \dot{u} \) is available so that

\[
\| \dot{W}_q(u) \|_{\infty} \leq k_A < \infty, \quad \| \dot{W}_q^*(u) \|_{\infty} < \infty
\]

where \( \| \cdot \|_{\infty} \) denotes the induced infinity norm of a time-varying matrix, and the matrix \( \dot{W}_q^*(u) : \mathbb{R}^n \rightarrow \mathbb{R}^m \) is defined by

\[
\dot{W}_q^*(u) = \dot{W}_q(u(t)) - \lambda_\nu W_q(u(t))
\]

and can be generated by the control system.

Remark: This assumption implies that the desired joint jerks are bounded and can be generated by the control system.

The modified DCAL is given by

\[
\tau = \dot{\dot{w}}_q(u) - F_p e_p - F_v \dot{e}_r - \dot{W}_q(u) \Gamma A W_q^T(u) z_p
\]

where \( F_p \) and \( F_v \) are positive definite gain matrices and \( \Gamma A \) is the adaptation gain.

The parametric adaptation law for generating the inverse dynamics function estimate \( \dot{w}_q(u) \) is now given by

\[
\dot{w}_q(u) = W_q(u) \dot{\Theta}, \quad \dot{\Theta} = \Gamma A W_q^T(u) z_p
\]

B. Repetitive Learning Control

Similar modifications to the ones described above are necessary to implement the repetitive learning law without using joint velocity signals.

**Assumption 5:** The desired trajectory vector \( u \) is sufficiently smooth so that a kernel \( K(t, \tau) \) which satisfies Assumption 2 can be constructed and in addition satisfies

\[
\int_0^T |\partial K(t, \tau)/\partial \tau|^2 \, d\tau < \infty.
\]

The modified repetitive learning control law is given by

\[
\tau = w_r - F_p e_p - F_v \dot{e}_r - k_L \Gamma L z_p
\]

where \( F_p \) and \( F_v \) are positive definite gain matrices, \( \Gamma L = \Gamma \Gamma L \) is the learning gain, and \( k_L \) was defined in (14).

The learning rule for generating the repetitive inverse dynamics function estimate \( \dot{w}_r(t) \) is now given by

\[
\dot{w}_r(t) = \int_0^T K(t, \tau) \partial K(t, \tau)/\partial \tau \, d\tau
\]

The kernel \( K^*(t, \tau) \) is given by

\[
K^*(t, \tau) = \partial K(t, \tau)/\partial \tau - \lambda_\nu K(t, \tau).
\]

The plot of the kernel \( K^*(0, \tau) \), when \( K(t, \tau) \) is a Gaussian kernel, is shown in Fig. 1. If the first partial derivatives of the Gaussian \( K(t, \tau) \) are discontinuous, Assumption 2 can be satisfied. However, in the experimental results, which will be presented in Section VI, we used a finite number of data points to generate both the kernels and the influence function estimate. Thus, these kernels have a finite eigenfunction expansion as detailed in Assumption 3.
IV. STABILITY ANALYSIS

In this section, we discuss the stability and convergence properties of the parametric adaptive and repetitive controllers presented in Section III. We will first analyze the repetitive controller in Section III-B. Subsequently, we will present stability and convergence results for the adaptive controller in Section III-A. Since the analysis of both schemes is almost identical, we will omit most of the details regarding the stability analysis of the parametric adaptive controller.

In order to facilitate our analysis, it is convenient to introduce the reference velocity estimation error to describe the estimation error dynamics, using the following linear transformation:

\[
\begin{bmatrix}
\hat{q}_p \\
\hat{q}_r
\end{bmatrix} =
\begin{bmatrix}
1 & 0 \\
\lambda_p & 1
\end{bmatrix}
\begin{bmatrix}
q_p \\
q_r
\end{bmatrix}
\]  
(34)

and to define the trajectory and observer error state \(e\) as follows:

\[
e = [e_p^T, e_r^T, \hat{e}_p^T, \hat{e}_r^T]^T.
\]  
(35)

A. Repetitive Control

In this section, we analyze the stability and convergence analysis of the repetitive control system introduced in Section III-B. Let us define the influence function error \(\hat{c}\) by

\[
\hat{c}(t, \sigma) = c(\sigma) - \hat{c}(t, \sigma).
\]  
(36)

The trajectory, observer, and learning error dynamics can then be represented as follows:

\[
\begin{align*}
e &= -\lambda_p e_p + e_r, \\
\dot{e}_r &= -\lambda_p \hat{q}_p + \hat{q}_r, \\
M(q) e_r &= -C(q, \dot{q}) e_r - \{\dot{F}_p + F_L\} e_p - F_v e_r \\
&- F_L \hat{q}_p + F_v \hat{q}_r - \hat{w}(u) - \Delta \hat{w}(u, e)
\end{align*}
\]  
(37)

\[
\begin{align*}
M(q) \dot{\hat{q}}_r &= -C(q, \dot{q}) \hat{q}_r - \{\dot{F}_p + F_L\} \hat{q}_p - \{\lambda_p \hat{M}(q)\} e_p \\
&- \{\dot{F}_{v} + \lambda_p \hat{M}(q)\} e_r - F_L \hat{q}_p \\
&- \dot{\hat{M}}(q) \hat{q}_r - \hat{w}(u) - \Delta \hat{w}(u, e) \\
&- C(q, \dot{q}) \{e_r - \hat{q}_r\}
\end{align*}
\]  
(38)

where \(F_L = k_L F_L, \Delta \hat{w}(u, e) = \hat{w}_L(q, \dot{q}, \hat{q}_p, \hat{q}_r) - \hat{w}_L(u), \hat{w}(u) = \hat{w}(u, \hat{c}) = w_L(u) - \hat{w}(u)\)

\[
\int_0^T K(t, \tau) \hat{c}(t, \tau) d\tau.
\]  
(42)

and

\[
\frac{\partial}{\partial t} \hat{c}(t, \tau) = -K^T(t, \tau) E_L \{e_p(t) + \hat{q}_p(t)\}
\]  
(43)

Theorem 1: Consider the system described by the error dynamics (37)–(43). For a given extended desired trajectory vector, \(u(t)\), if Assumptions 2 and 5 are satisfied, and \(\hat{q}_p(0) = 0\):

\[
\begin{align*}
|e_p(0)|^2 &\leq c_1, \\
|e_r(0)|^2 &\leq c_2, \\
|\hat{q}_r(0)|^2 &\leq c_3
\end{align*}
\]  
(44)

\[
\int_0^T |F_L^{-1/2} \hat{c}(0, \gamma)|^2 d\gamma \leq c_4
\]  
(44)

it is always possible to choose feedback gains \(F_p, F_v, \lambda_p, K\), and the observer gain \(\lambda_o\) so that the origin of the system (37)–(43) is locally uniformly stable and \(\hat{q}_p(t) \to 0, \forall t \in [0, T]\).

2) If, in addition, the finite dimensionality Assumption 3 is satisfied, then the origin of the state space, \((e, \hat{c}(\gamma)) = 0, \forall \gamma \in [0, T]\), is locally uniformly exponentially stable.

Remarks: Since \(q\) is measurable, we can always set \(\hat{q}_p(0) = 0\). Part 2) of Theorem 1 guarantees that the repetitive learning control system has a certain degree of robustness to unmodeled disturbance inputs [16]. This in turn provides robustness to discretization errors in actual implementations, where the functions, \(\hat{w}_L(\cdot, \hat{c}), \hat{c}(\cdot, \cdot)\) are discretized into finite elements.
Proof: We will only prove part 1) of this theorem. The proof of part 2) is very similar to the analysis presented in [17] and will be omitted.

Define the Lyapunov functional candidate \( V(t, \bar{e}, \bar{\bar{e}}) \) as follows:

\[
V(t, \bar{e}, \bar{\bar{e}}) = V_1(t, \bar{e}) + V_2(t, \bar{e}, \bar{\bar{e}})
\]

(45)

\[
V_1(t, \bar{e}) = \frac{1}{2} \left( e_t^T \Phi_f + \lambda_1^2 \sum_{j=1}^n e_j \bar{e}_j + \tilde{q}_r^T Q \tilde{q}_r + e_t^T \Theta \bar{e}_r \right)
\]

(46)

\[
V_2(t, \bar{e}, \bar{\bar{e}}) = \frac{1}{2} \int_0^T \left[ K(t, \bar{\bar{e}})(\bar{e}_r + \tilde{q}_r) + \tilde{\Gamma}^T \tilde{\Gamma} / 2 \right] d\gamma
\]

(47)

where

\[
\left( \tilde{\Gamma}^T / 2 \right)^2 = \Gamma_L, \quad \lambda = \frac{1}{2}(\lambda_M + \lambda_m).
\]

\( \lambda_M \) and \( \lambda_m \) are maximum and minimum eigenvalues of \( M(q) \) and \( Q \) is a positive definite matrix to be defined subsequently. In the remainder of this section, we will sometimes denote \( M = M(q) \) in the interest of compactness.

Remark: It is straightforward to show that \( V(t, \bar{e}, \bar{\bar{e}}) \) is a positive definite functional.

Differentiating \( V_2 \) with respect to time, utilizing (37), (38), and the learning law (32), we obtain

\[
\dot{V}_2 = \int_0^T \left[ \left( \tilde{\Gamma}^T / 2 \right) \bar{K}(t, \bar{\bar{e}})(\bar{e}_r + \tilde{q}_r) + \tilde{\Gamma}^T / 2 \tilde{\Gamma} \right] d\gamma
\]

(48)

The first term in (48) will be cancelled by the last term of the control law (30), while the second term in (48) cancels the effect of term \( \tilde{\Gamma} \) in the error dynamics (39) and (40).

Differentiating \( V_2 \) with respect to time, utilizing the skew symmetric property of the matrix \( [\tilde{\Gamma}(q) - 2C(q, \dot{q})] \) and combining this result with (48), we obtain

\[
\dot{V} = -\lambda_2 \bar{e}_t^T \Phi_f + \lambda_2 \sum_{j=1}^n e_j \bar{e}_j - \lambda \bar{q}_r^T Q \tilde{q}_r - \bar{e}_t^T F \bar{e}_r
\]

(49)

As shown in Appendix A, the terms \( e_t^T + \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r \) in (49) satisfy the following inequality:

\[
-(\bar{e}_t^T + \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r) \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r \leq (b_1 \bar{q}_r, \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r) + \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r
\]

(50)

where

\[
b_1(\bar{q}_r, \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r) + \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{\Gamma} \tilde{e}_r
\]
Thus, \( \dot{\theta} \in L_{\infty} \), which in turn implies that \( \dot{e} \in L_{\infty} \). The asymptotic convergence to zero of \( e(t) \) follows from Barbalat’s Lemma.

**B. Adaptive Control**

In this section, we analyze the stability and convergence analysis of the adaptive control system introduced in Section III-A. Let us define the parameter error \( \dot{\theta} \) by

\[
\dot{\theta}(t) = \theta - \dot{\theta}(t).
\]

Utilizing the definition of error state \( e \) given by (35), the trajectory, observer, and adaptive control error dynamics can then be represented by (37)–(41), where \( F_L = W_q(u)\Gamma^T \tilde{q}(u) \):

\[
\dot{u}(u) = \dot{u}(u, \dot{\theta}) = \dot{q}(u) - \dot{u}(u) = W_q(u)\dot{\theta}
\]

and

\[
\dot{\theta}(t) = -\Gamma^T \tilde{q}(t)\{e_p(t) + \tilde{q}_p(t)\}.
\]

**Theorem 2:** Consider the system described by the error dynamics (37)–(41) and (60)–(61). For a given extended desired trajectory vector, \( \dot{u}(t) \), if Assumption 4 is satisfied, and

\[
|e_p(0)|^2 \leq \epsilon_1, \quad |e_v(0)|^2 \leq \epsilon_2, \quad |\tilde{q}_p(0)|^2 \leq \epsilon_3
\]

\[
|F^{-1/2}_{\theta}(0)|^2 \leq \epsilon_4
\]

it is always possible to choose feedback gains \( F_{pp}, F_{vv}, \lambda_p \) and the observer gain \( \Lambda_0 \) so that the origin of the state space, \( (e, \dot{\theta}) = 0 \), is locally uniformly stable and \( \lim_{t \to \infty} \dot{V}(t) = 0 \).

Proof: We will only proof part 1) of this theorem. The proof of part 2) is very similar to the analysis presented in [17] and will be omitted. Define the Lyapunov function candidate

\[
V(t, e, \dot{\theta}) = V_1(t, e) + V_2(t, e, \dot{\theta})
\]

with \( V_1 \) given by (46) and

\[
V_2 = \frac{1}{2} \Gamma^{-1/2} W_q(u(t))\{e_p + \tilde{q}_p\} + \Gamma^{-1/2} e(0)^2.
\]

Differentiating \( V_2 \) with respect to time, utilizing (37), (38), and (61), we obtain

\[
\dot{V}_2 = [\Gamma^{-1/2} W_q(u(t))\{e_p + \tilde{q}_p\} + \Gamma^{-1/2} e(0)^2]
\]

\[
\cdot [\Gamma^{-1/2} W_q(u(t))\{e_p + \tilde{q}_p\} + \Gamma^{-1/2} e(0)^2]
\]

\[
+ \Gamma^{-1/2} e(0)^2 = (e_r + \tilde{q}_p)^T W_q^T(u)\Gamma A_W q(u)\{e_p + \tilde{q}_p\}
\]

\[
+ (e_r + \tilde{q}_p)^T \dot{u}(u, \dot{\theta}).
\]

The first term in (65) will be cancelled by the last term of the control law (28), while the second term in (65) cancels the effect of term \( \dot{u}(u, \dot{\theta}) \) in the error dynamics (39) and (40).

Differentiating \( V_1 \) with respect to time and combining this result with (65), we obtain (49). The rest of the proof is the same as the proof of Theorem 1.

**V. DELAYED LEARNING RULE**

In order to implement the learning algorithm described in (31) and (32) in real time, it is necessary to utilize a machine with massive parallel processing capabilities such as a neural network, since both the inverse dynamics estimate and the influence function estimate must be updated in parallel. However, in the experimental results which will be presented subsequently, a digital personal computer was used to implement the controller. Thus, in this case, the influence function estimates cannot be updated continuously and must be constant during a certain period of time, which is sufficiently large for the update algorithm to be computed.

To successfully implement the repetitive learning algorithm using a conventional digital computer, an algorithm similar to the delayed learning rule introduced in [10] should be formulated. We now introduce a modified version of the learning algorithm in (31) and (32), in which the influence function \( \tilde{\alpha}_k(\gamma) \) is updated at \( \Delta T \) time intervals.

The delayed learning rule for generating the inverse dynamic function estimate \( \dot{u}(t) \) in (30) is given by

\[
\dot{u}(t) = \int_0^T K(t, \tau) e(\tau) d\tau
\]

\[
\tilde{\alpha}(\gamma) = \tilde{\alpha}_{k-1}(\gamma) + \int_{(i-1)\Delta T}^{i\Delta T} K^\nu(\sigma, \tau) \Gamma_L z_p(\tau) d\sigma - \Gamma_L
\]

\[
\{K(\Delta T, \tau) \Gamma_L z_p(k) - K((k-1)\Delta T, \tau) \Gamma_L z_p(k-1)\}
\]

for \( k\Delta T \leq t < (k+1)\Delta T \). The auxiliary error signal \( z_p \) was defined in (25) and \( z_p(k) \) denotes \( z_p(k\Delta T) \).

In this algorithm, the influence function \( \tilde{\alpha}(\gamma) \) is only updated at discrete time intervals (i.e., \( t = k\Delta T, k = 0, 1, 2, \ldots \)). It remains unchanged for \( (k-1)\Delta T \leq t < k\Delta T \). Notice that in many repetitive control applications we can set \( \Delta T = T \) (i.e., the computational delay is equal to one full repetitive cycle).

**Theorem 3:** Consider the system described by the error dynamics in (37)–(40) and the adaptation law in (66) and (67). Under the same conditions of Theorem 1, given the bounds (44) it is always possible to choose feedback gains \( F_{pp}, F_{vv}, \lambda_p \) and the observer gain \( \Lambda_0 \) so that the origin of the system (37)–(42) is locally uniformly stable and \( \lim_{t \to \infty} e(0) = 0 \).

Proof: Consider the discrete time Lyapunov functional candidate defined by

\[
V_k = V_1(k\Delta T, e(k\Delta T)) + V_4(k\Delta T, e(k\Delta T), \tilde{\alpha}(\gamma))
\]

where \( e \) was defined in (35), \( V_1(t, e) \) is defined in (46), and

\[
V_4 = \frac{1}{2} \int_0^T \tilde{\alpha}_k(\gamma) \Gamma_L - \frac{1}{2} k\Delta T \tilde{\alpha}_k(\gamma) + \frac{1}{2} k\Delta T \tilde{\alpha}_k(\gamma) \Gamma_L z_p(t) d\gamma
\]

\[
\tilde{\alpha}_k(\gamma) = e(\gamma) - \tilde{\alpha}_k(\gamma).
\]

It is straightforward to show that \( V_k \) is a positive definite functional.
Let us calculate $\Delta V_k$ by
\[
\Delta V_k = V_k - V_{k-1} = \int_{(k-1)\Delta T}^{k\Delta T} \dot{V}_1(\tau) d\tau + V_{d(k)} - V_{d(k-1)},
\] (70)

From (45)–(47) and (53), and using (49), we obtain
\[
\dot{V}_1(t, e) \leq -e_o^T Q_o e_o - k_L z_p^T(t) \Gamma_L z_p(t) - z_p^T(t) \dot{\varphi}(u(t), \bar{e})
\] (71)

where
\[
z_p(t) = e_p(t) + \dot{e}_p(t) = \dot{z}_p(t) + \lambda_p z_p(t)
\] (72)
and $e_o$ is defined in (54).

Integrating the second term in (71), utilizing (25) and (72), we obtain
\[
\int_{(k-1)\Delta T}^{k\Delta T} k_L z_p^T(\tau) \Gamma_L z_p(\tau) d\tau = \frac{1}{2} k_L z_p^T(t) \Gamma_L z_p(t) \int_{(k-1)\Delta T}^{k\Delta T} \Gamma_L z_p(\tau) d\tau + \frac{1}{2} k_L \Delta T z_p^T(t) \Gamma_L z_p(t).
\] (73)

Noting that
\[
\bar{e}(t, \tau) = \bar{e}_{k-1}(\tau) \text{ for } (k-1)\Delta T \leq t < k\Delta T
\] (74)
then, from (68) through (73), we obtain
\[
\Delta V_k \leq -\int_{(k-1)\Delta T}^{k\Delta T} e_o^T Q_o e_o d\tau + \int_{(k-1)\Delta T}^{k\Delta T} z_p^T(t) \\
\cdot \dot{\varphi}(u(t), \bar{e}_{k-1}(\tau)) d\tau - \int_{(k-1)\Delta T}^{k\Delta T} k_L z_p^T(\tau) \Gamma_L z_p(\tau) d\tau + \frac{1}{2} k_L z_p^T(t) \Gamma_L z_p(t) \int_{(k-1)\Delta T}^{k\Delta T} \Gamma_L z_p(\tau) d\tau
\]
\[
+ \frac{1}{2} \int_{(k-1)\Delta T}^{k\Delta T} \{e_o^T(\tau) \Gamma_L^{-1} \bar{e}_o(\tau) - e_o^T(\tau) \Gamma_L^{-1} \bar{e}_{k-1}(\tau) \}
\cdot \bar{e}_{k-1}(\tau) d\tau
\] (75)

Using (33), (67), (69), and (72), $\bar{e}_k(\tau)$ can be expressed as
\[
\bar{e}_k(\tau) = \bar{e}_{k-1}(\gamma) + \int_{(k-1)\Delta T}^{k\Delta T} K(\tau, \gamma) \Gamma_L z_p(\tau) d\tau.
\] (76)

Therefore, in (75)
\[
\frac{1}{2} \int_{0}^{T} \{e_o^T(\tau) \Gamma_L^{-1} \bar{e}_o(\gamma) - e_o^T(\tau) \Gamma_L^{-1} \bar{e}_{k-1}(\gamma) \}
\]
\[
= \frac{1}{2} \int_{(k-1)\Delta T}^{k\Delta T} z_p^T(\tau) \dot{\varphi}(u(t), \bar{e}_{k-1}(\tau)) d\tau
\]
\[
+ \frac{1}{2} \int_{0}^{T} \Gamma_L^{1/2} \int_{(k-1)\Delta T}^{k\Delta T} K(\tau, \gamma) z_p(\tau) d\tau d\gamma.
\] (77)

Applying the Schwartz inequality to the last term of (77), we obtain
\[
\frac{1}{2} \int_{0}^{T} \Gamma_L^{1/2} \int_{(k-1)\Delta T}^{k\Delta T} K(\tau, \gamma) z_p(\tau) d\tau d\gamma
\leq \frac{1}{2} k_L \Delta T \int_{(k-1)\Delta T}^{k\Delta T} z_p^T(\tau) \Gamma_L z_p(\tau) d\tau.
\] (78)

Now, combining (75), (77), and (78), we obtain the following inequality:
\[
\Delta V_k \leq -\int_{(k-1)\Delta T}^{k\Delta T} e_o^T Q_o e_o d\tau + \frac{1}{2} k_L \Delta T \int_{(k-1)\Delta T}^{k\Delta T} z_p^T(\tau) \Gamma_L z_p(\tau) d\tau.
\] (79)

By choosing $\Gamma_L = \sigma_I I$, and using (25) and (72) we obtain
\[
\frac{1}{2} k_L \Delta T z_p^T(t) \Gamma_L z_p(t)
\leq \frac{1}{2} k_L 1 \Delta T z_p^T(t) \Gamma_L z_p(t)
\leq \frac{1}{2} k_L \sigma I \Delta T \{e_o^2 + \dot{e}_o^2 + 2\dot{e}_o \mid \dot{e}_o \mid \}
\] (80)
and the following final inequality:
\[
\Delta V_k \leq -\int_{(k-1)\Delta T}^{k\Delta T} e_o^T Q_o - Q_o d\tau
\] (81)

where $Q_o$ is defined in (55) and $\Delta Q$ is given by
\[
\Delta Q = \frac{1}{2} k_L \sigma I \Delta T \left[ \begin{array}{cccc}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 \\
0 & 0 & 1 & 1 \\
\end{array} \right].
\]
As shown in Appendix C, given the bounds (44), it is always possible to choose gains $\lambda_p, \sigma_p, \sigma_q, \sigma_q$, and $\lambda_0$ such that $\sigma_o - \Delta T > 0$. The rest of the proof is same as that of Theorem 1.

VI. SIMULATION AND IMPLEMENTATION RESULTS

A simulation study using the dynamic model of the Berkeley/NSK SCARA two-axis manipulator was conducted to test the performance of the repetitive learning control law in Section III-B and the delayed repetitive learning rule in Section V. Subsequently, the delayed repetitive learning rule in Section V was implemented on the Berkeley/NSK SCARA two-axis manipulator. In this section, we describe some of the results obtained in this study. A detailed description of the Berkeley/NSK arm and the model employed in the simulation study can be found in [18].

The periodic desired trajectories used in the simulations are plotted in Fig. 2. Notice that the desired position, velocity, and acceleration were generated so that Assumption 5, which relates to the smoothness of the desired acceleration, is satisfied. The kernel $K(\tau, \gamma)$ used in the simulations was generated using a Gaussian distribution function:
\[
K(\tau, \gamma) = f(\tau - \gamma - kT) \text{ for } |\tau - \gamma| \leq kT - \frac{T}{2}
\]
\[
f(x) = \frac{1}{\sqrt{2\pi}} \exp(-x^2/2\sigma^2)
\] (82)

and the extended kernel $K^*(\tau, \gamma)$ was calculated using (33).
Fig. 2. The desired trajectories.

$K(0, \tau)$ and $K^* (0, \tau)$ are plotted in Fig. 1. Notice that the width of Gaussian distribution in Fig. 1 can be adjusted by changing $\sigma$ in (82). A value of $\sigma = 0.04$ was chosen in both the simulation and experimental studies.

A. Learning Control with Velocity Estimation

The repetitive learning rule described by (31)–(32) was simulated assuming that the estimates of both the influence function and inverse dynamic function are updated simultaneously and continuously.

In the simulation study, the Euler integration method was employed to solve the robot dynamic equations (1) and observer differential equations (21)–(22). The integration step size was 0.0002 s. The influence function estimate $\hat{\chi}(t, \tau)$ and kernels $K(t, \tau), K^*(t, \tau)$ were discretized into an array of 3200 finite elements. The inverse dynamics function estimate $\hat{\theta}_r(\cdot)$ defined in (31) was obtained by numerical quadrature, while the influence function estimate $\hat{\chi}(\cdot, \cdot)$ was updated using the Euler numerical integration method.

Fig. 3 shows the simulation result of the position tracking error $e_{p1}(t)$ for the first axis. In the figures, the upper plot shows the results when the repetitive learning control rule defined by (30)–(32) is used, while the lower plots show the corresponding results when the conventional learning rule in [10] is used, assuming that direct measurement of the joint velocities is possible. In these simulations, the desired trajectory shown in Fig. 2 is repeated every 5 s.
Notice that, due to the fact that relatively high gains are used in the simulations, the position tracking errors are very similar in both cases. In the beginning, however, the errors in the upper plot are slightly larger than those of the lower plot. The velocity estimation error of the observer is also plotted in Fig. 4. Since the observer is constructed based on simple desired input/output relations, the observer cannot accurately estimate the actual velocities at the beginning of the learning cycle. As learning proceeds, however, the observer errors converge to zero as well as tracking errors.

These simulation results support the theoretical results obtained in the previous sections, mainly that the presented learning rule is locally exponentially stable, and that the observer errors, learning error, and tracking error signals converge to zero.

B. Delayed Learning Control with Velocity Estimation

The delayed learning rule described by (66)–(67) was tested by both simulation and experimental studies. The observer defined by the differential equations (21)–(22) was digitally implemented using the transition matrix of (21) and (22), in both the simulation and implementation studies. The influence function estimate $\hat{c}_s(t)$ was updated by using (32) only at the beginning of every repetitive cycle.

Fig. 5 shows the simulation results of the position tracking error, $e_{3d}(t)$, for the first axis. In the figure, the upper plot shows the results when the delayed repetitive learning rule defined by (66)–(67) is used, while the lower plots show the corresponding results when the conventional delayed learning rule in [10] is used, assuming that the direct measurements of joint velocities are available. It is interesting to note that in Fig. 5 the position tracking errors at the beginning of the upper plot are smaller than those of the lower plot. The observer-based learning rule presented in this paper utilizes an additional position feedback term. This means that if the same position error exists, the learning rule will exhibit a larger position feedback action than the conventional learning rule. The velocity estimation error of the observer is also plotted in Fig. 6. From these figures, it can be concluded that the convergence of the new method is somewhat slower than that of the conventional method. As learning proceeds, however, the observer errors as well as tracking errors converge to zero. Thus, it is verified that the proposed scheme is successful when the velocity vector is not measurable.

The delayed repetitive control algorithm was implemented on the Berkeley/NSK manipulator using an IBM PC/AT as the controller. A detailed description of the experimental setup can be found in [10]. It should be emphasized that the on-line
computational complexity of the control law (30) is not much more significant than that of a simple linear-state variable feedback action with linear-state estimation. The first term in (30), \( \dot{\mathbf{u}}_p \), can be computed off-line and stored at the beginning of each learning cycle. The remaining feedback terms in (30) are those corresponding to the state variable feedback action. The amount of off-line calculations involved in computing the learning laws given by (66) and (67) is not as large as it may appear. Notice that, by selecting a kernel with a sufficiently small support, e.g., selecting a sufficiently small variance \( \sigma \) for the Gaussian kernel in (82), most of the elements of the discretized kernel \( K(t, \tau) \) will be zero. Thus, only a relatively small number of multiplications and additions need to be performed in the computation of \( \dot{\mathbf{u}}_p(t) \) given by (66) and \( \dot{\mathbf{q}}_p(t) \) given by (67). For example, by choosing \( \sigma = 0.04 \) in our experimental study result, \( f(x) \) given by (82) is very small for \( |x| > 0.32 \). Consequently, \( K(t, \tau) \) and \( K^*(t, \tau) \) were set to zero for \( |t - \tau| > 0.32 \) s. Thus, based on a sampling time of 4 ms, 160 multiplications and additions are needed for updating each element of \( \dot{\mathbf{u}}_p(t) \) in (66).

Since we were using a relatively slow IBM PC/AT in our experiments, a pause of about 30 s was inserted in the desired trajectory between each repetitive cycle, in order to update the influence function and inverse dynamics function estimates. This period is not shown in Fig. 7. This off-line computational time can be significantly reduced using a more powerful processor.

The upper plot in Fig. 7 shows the position tracking error when the observer-based learning method presented in this paper was used. The lower plot shows the corresponding results when the original learning method in [10] is used and the velocity signals are obtained by simple numerical differentiation. As shown by the figures, almost the same results were obtained. Notice, however, that due to the fact that the velocity signal which is obtained by numerical differentiation is noisy, the error signals in the conventional learning algorithm are noisier than the error signals in the learning algorithm with velocity estimation. These results confirm the stability and usefulness of the new learning algorithm.

**VII. CONCLUSION**

New repetitive and adaptive control schemes for robot manipulators with velocity estimation were presented in this paper. The proposed observer-based control schemes do not require the direct measurement of the joint velocity vector, as is necessary in previous adaptive and learning schemes.
In the case of repetitive control, the unknown inverse dynamic function of the robot manipulator was represented by an integral equation of the first kind, utilizing the ideas in [10]. A simple linear-state observer was introduced to obtain estimates of the joint velocity signals. The error signal used in the adaptation and learning algorithm is a linear combination of the position tracking and estimation error signals. The local exponential stability of the proposed scheme is rigorously proven. An observer-based delayed repetitive learning rule was also presented, which is useful in real-time implementations. Simulation and experimental results utilizing the Berkeley/NSK SCARA robot show that the proposed schemes are useful when the joint velocity vector is not measurable.

**APPENDIX A**

**DERIVATION OF (50)**

The derivation of (50) is very similar to the proof of [9, Lemma 1]. From (41), applying [9, Lemma 1], we obtain

\[-(c^T + \hat{q}^T) T \Delta \omega - (c^T + \hat{q}^T) C(q, \dot{q}) \{ e_r - \hat{q}_r \} \]

\[\leq - (c^T + \hat{q}^T) (\lambda_p M \dot{e}_p - \lambda_p M e_r) + b_{2,2} (\hat{q}_d, \hat{q}_d) \{ e_p + |\dot{q}_r| \} + b_{2,3} \lambda_p |e_p| + b_{2,3} |e_p|^2 \]

\[+ \{ b_{1,1} (\hat{q}_d, \dot{q}_d) + 2 \lambda_p b_{2,2} (\hat{q}_d) \} |e_p| |e_r| + |\dot{q}_r| \]

\[+ (2 b_{2,2} (\dot{q}_d) + b_{2,3}) |e_p||e_r| + b_{2,3} |\dot{q}_r| \]

\[+ b_{2,3} |\dot{q}_r|^2 + \lambda_p b_{2,3} |e_p|^2 \]  \hspace{1cm} (A3)

where \( \hat{b}_{1,1} = b_{1,1} \cdot (|e_r| + |\dot{q}_r|) \).

Let us define \( \hat{b}_2 (e_r, \dot{q}_r) \) by

\[\hat{b}_2 (e_r, \dot{q}_r) = 2 b_{2,2} (\dot{q}_d) + b_{2,3} \cdot (|e_r| + |\dot{q}_r|).\] \hspace{1cm} (A4)

Then, by adding up (A1) and (A2), and noting that \( \hat{b}_2 (e_r, \dot{q}_r) > \hat{b}_{1,1} \cdot (|e_r| + |\dot{q}_r|) \), we can obtain (50).

**APPENDIX B**

**SUFFICIENT CONDITIONS FOR $Q_o > 0$**

Here we derive sufficient conditions for $Q_o$ in (53) to be positive definite. Using Sylvester’s theorem, $Q_o$ as given by (55) is positive definite if the following inequalities are satisfied:

\[Q_{11} > 2 \max \{ Q_{13}, Q_{24} \}, \quad Q_{22} > 2 Q_{24} \]

\[Q_{33} > 2 \max \{ Q_{13}, Q_{23} \}, \quad Q_{44} > 2 \max \{ Q_{14}, Q_{24}, Q_{34} \} \]

\[Q_{13} Q_{24} - Q_{14} Q_{23} > 0.\]

From the above inequalities, using (56) and performing some algebra, we obtain the following conditions:

\[\sigma_p > \max \left[ \frac{\lambda_p^2 \lambda_m - \lambda_m}{2}, \frac{b_{1,1} (\dot{q}_d, \dot{q}_d) + \lambda_p (\lambda_p + 1) \hat{b}_2 (e_r, \dot{q}_r) - \lambda_p^2 \lambda_m}{\lambda_p - 1} \right] \]

\[\lambda_p > 1 \]

\[\sigma_q > b_{1,1} (\dot{q}_d, \dot{q}_d) + (\lambda_p + 1) \left\{ \hat{b}_2 (e_r, \dot{q}_r) + \lambda_p M - \lambda_m \right\} \]

\[\sigma_q > 4 (\sigma_p + b_{1,1} (\dot{q}_d, \dot{q}_d) + \lambda_p b_2 (e_r, \dot{q}_r)) \]

\[\lambda_q > \sigma_q + \lambda_p \lambda_m + b_2 (e_r, \dot{q}_r).\] \hspace{1cm} (B1)

Notice that most of the inequalities in (B1) contain the term \( \hat{b}_2 (e_r, \dot{q}_r) \) which was defined in (A4). For any given $\tau_1$ such that

\[|e_r(t)| + |\dot{q}_r(t)| \leq \tau_1, \quad t \geq 0 \] \hspace{1cm} (B.2)

the following is true:

\[\hat{b}_2 (e_r, \dot{q}_r) \leq \hat{b}_2 (\tau_1) = 2 b_{2,2} (\dot{q}_d) + b_{2,3} \tau_1\] \hspace{1cm} (B3)

where \( b_{1,1} (\dot{q}_d, \dot{q}_d), b_{2,2} (\dot{q}_d), b_{2,3} \) are positive constants once the desired velocity and acceleration are specified.

We will now obtain an expression for the constant $\tau_1$ in (B2), in terms of the bounds (44). Notice that, since $q$ is measurable, we can always set $|\dot{q}_p(0)| = 0$. 

Using the mean value theorem (MVT) and similar algebraic manipulations as in [9], we obtain

\[-(c^T + \hat{q}^T) (e_r - \dot{q}_r) \]

\[\leq \{ b_{1,1} (\dot{q}_d, \dot{q}_d) + 2 \lambda_p b_{2,2} (\dot{q}_d) \} |e_p| |e_r| + |\dot{q}_r| \]

\[+ b_{2,2} (\dot{q}_d) |e_p| + b_{2,3} |e_p||e_r| + \lambda_p b_{2,3} |e_p|^2 \]

\[+ \lambda_p b_{2,3} |\dot{q}_r|^2 + \lambda_p b_{2,3} |e_p|^2 \] \hspace{1cm} (A2)
By (45)–(47) and (53):
\[
\frac{1}{2} \lambda_m \| e_\tau \|^2 \leq V(t, e, \dot{e}) \leq V(0, e(0), \dot{e}(0))
\]
\[
\frac{1}{2} \lambda_m \| \dot{g}_\tau \|^2 \leq V(t, e, \dot{e}) \leq V(0, e(0), \dot{e}(0))
\]
and
\[
| e_k | + | \dot{g}_k | \leq \sqrt{\frac{8V(0, e(0), \dot{e}(0))}{\lambda_m}}
\]
where
\[
V(0, e(0), \dot{e}(0)) \leq \frac{1}{2} \left\{ (\sigma_p + \lambda_p \| \dot{g}_k \|^2 + k_L \| \Gamma_L \|) e_1 + \lambda_M (e_2 + e_3 + e_4) \right\}
\]
Thus, \( r_1 \) can be defined as follows:
\[
r_1 = 4/\lambda_M [(\sigma_p + \lambda_p \| \dot{g}_k \|^2 + k_L \| \Gamma_L \|) e_1 + \lambda_M (e_2 + e_3 + e_4)]
\]
Substituting (B7) and (B3) into the first inequality in (B1), we obtain the following sufficient condition which the gain \( \sigma_p \) must satisfy:
\[
\omega \sigma_p^2 + b \sigma_p + c > 0
\]
where \( \lambda_p > 1 \).

\[
a = (\lambda_p - 1)^2
\]
\[
b = (\lambda_p + 1)(\lambda_p^2 \lambda_p - b_1(\dot{q}_k, \dot{d}_k) - 2 \lambda_p (\lambda_p + 1)b_2(\dot{q}_k))
\]
\[
c = (\lambda_p^3 \lambda_p - b_1(\dot{q}_k, \dot{d}_k) - 2 \lambda_p (\lambda_p + 1)b_2(\dot{q}_k))^2
\]
\[
- (4/\lambda_M)^2 \lambda_p^2 (\lambda_p + 1)^2 b_2^2 e_1 + \lambda_M (e_2 + e_3 + e_4)
\]
It is clear that a sufficiently large gain \( \sigma_p \) that will satisfy (B8) can always be found. Likewise, sufficiently large gains \( \sigma_{p1}, \sigma_{p2} \) and \( \lambda_0 \) that will satisfy the remaining inequalities in (B1) can always be found.

**APPENDIX C**

Using \( \hat{b}_2(e_\tau, \dot{q}_\tau) \) defined in (A4), let us define \( \hat{b}_2(e_\tau, \dot{q}_\tau, \sigma_L \Delta T) \) as follows:
\[
\hat{b}_2(e_\tau, \dot{q}_\tau, \sigma_L \Delta T) = \hat{b}_2(e_\tau, \dot{q}_\tau) + \Delta T k_L \sigma_L.
\]
Then, \( Q_o \) can be expressed as
\[
Q_o(\hat{b}_2(e_\tau, \dot{q}_\tau)) = Q_o(\hat{b}_2(e_\tau, \dot{q}_\tau, \sigma_L \Delta T)) + \bar{Q}
\]
where
\[
\bar{Q} = \frac{1}{2} \Delta T k_L \sigma_L \begin{bmatrix}
2 \lambda_p^2 & 0 & \lambda_p & \lambda_p \\
0 & 0 & 0 & 0 \\
\lambda_p & 0 & 2 & 1 \\
\lambda_p & 0 & 1 & 2
\end{bmatrix} \geq 0.
\]
Using (C2), \( Q_o - \Delta Q \) in (81) can be expressed as
\[
Q_o - \Delta Q = Q_o(\hat{b}_2(e_\tau, \dot{q}_\tau, \sigma_L \Delta T)) + \bar{Q} - \Delta Q.
\]

Note that, from (82),
\[
\bar{Q} - \Delta Q = \frac{1}{2} \Delta T k_L \sigma_L \begin{bmatrix}
2 \lambda_p^2 & 0 & \lambda_p & \lambda_p \\
0 & 0 & 0 & 0 \\
\lambda_p & 0 & 1 & 0 \\
\lambda_p & 0 & 0 & 1
\end{bmatrix} \geq 0.
\]
Therefore, in order to guarantee that \( Q_o - \Delta Q > 0 \), it is sufficient that
\[
Q_o(\hat{b}_2(e_\tau, \dot{q}_\tau, \sigma_L \Delta T)) > 0.
\]
Thus, the sufficient conditions which the constants \( \lambda_p, \sigma_{p1}, \sigma_{p2}, \lambda_0 \) and \( \lambda_0 \) must satisfy in order for \( Q_o - \Delta Q > 0 \) are obtained in the same manner as the sufficient conditions for \( \bar{Q} > 0 \) derived in Appendix B, except that \( \hat{b}_2(e_\tau, \dot{q}_\tau) \) should be replaced by \( \hat{b}_2(e_\tau, \dot{q}_\tau, \sigma_L \Delta T) \).

**REFERENCES**


Kazumasa Kaneko was born in Tokyo, Japan, in 1957. He received the B.S. and M.S. degrees in control engineering from the Tokyo Institute of Technology in 1980 and 1982, respectively. He joined the Nippon Telegraph and Telephone Corporation (NTT), Tokyo, Japan, in 1982 and has been engaged in research and development on mechatronics systems for mass storage systems and telecommunication systems. In 1991, he was a Visiting Industrial Fellow with the Department of Mechanical Engineering, University of California at Berkeley. He is currently a Senior Research Engineer at NTT Opto-electronics Laboratories.

Mr. Kaneko is a member of The Society of Instrument and Control Engineers in Japan, The Japan Society of Mechanical Engineers, and The Japan Society for Precision Engineering. He received the Young Authors Award from The Society of Instrument and Control Engineers in 1980, and the JSME Medal from The Japan Society of Mechanical Engineers in 1993.

Roberto Horowitz (M’89) was born in Caracas, Venezuela, in 1955. He received the B.S. degree with highest honors in 1978 and the Ph.D. degree in 1983 in mechanical engineering from the University of California at Berkeley. In 1982, he joined the Department of Mechanical Engineering at the University of California at Berkeley, where he is currently a Professor. He teaches and conducts research in the areas of adaptive, learning, nonlinear and optimal control, with applications to micro-electro-mechanical systems (MEMS), computer disk file systems, robotics, mechatronics, and intelligent vehicle and highway systems (IVHS).

Dr. Horowitz is a member of ASME. He was a recipient of the 1984 IBM Young Faculty Development Award and the 1987 National Science Foundation Presidential Young Investigator Award.