Available online at www.sciencedirect.com
SCIENCE DIRECT®
Mechatronics 16 (2006) 51–61
MECHATRONICS

An adaptive switching learning control method for trajectory tracking of robot manipulators

P.R. Ouyang [a], W.J. Zhang [a,*], Madan M. Gupta [b]

[a] *Advanced Engineering Design Laboratory, Department of Mechanical Engineering, University of Saskatchewan, 57 Campus Drive, Saskatoon, SK, Canada S7N 5A9*
[b] *Intelligent Systems Research Laboratory, Department of Mechanical Engineering, University of Saskatchewan, 57 Campus Drive, Saskatoon, SK, Canada S7N 5A9*

Received 21 March 2004; accepted 2 August 2005

Abstract

In this paper, a new adaptive switching learning control approach, called adaptive switching learning PD control (ASL-PD), is proposed for trajectory tracking of robot manipulators in an iterative operation mode. The ASL-PD control method is a combination of the feedback PD control law with a gain switching technique and the feedforward learning control law with the input torque profile. The torque profile is updated by the previous torque profile (which makes sense for learning). Furthermore, in this new control method, the switching control scheme is integrated into the iterative learning procedure; as such, the trajectory tracking converges very fast. The ASL-PD method achieves the asymptotical convergence based on the Lyapunov's method. The ASL-PD method possesses both adaptive and learning capabilities with a simple control structure. The simulation study validates this new method. In particular, both position and velocity tracking errors monotonically decrease with the increase of the number of iterations. The convergence rate with the ASL-PD method is faster than that of the adaptive iterative learning control method proposed by others in literature.

Keywords: Adaptive control; Iterative learning; Switching gain control; PD control; Trajectory tracking; Robot manipulator

1. Introduction

The control of robot manipulators has attracted a great deal of attentions due to their complex dynamics and wide applications in industrial systems. Basically, the control methods can be classified into the following three types. The first type is the traditional feedback control (proportional–integral–derivative (PID) control or proportional–derivative (PD) control [1–4]) where the errors between the desired and the actual performance are treated in certain ways (proportional, derivative, and integral), multiplied by gains, and fed back as the "correct" input torque. The second type is the adaptive control [5–11] where the controller modifies its behaviour in response to the changes in the dynamics of the robot manipulator and the characteristics of the disturbances received by the manipulator system. The third type is the iterative learning control (ILC) [12–17] where the previous torque profile is added to the current torque in a certain manner. Some other control methods, including the robust control, model based control, switching control, and sliding mode control, can be in one or another way reviewed either as specialization and/or combination of the three basic types, or are simply different names due to different emphases when the basic types are examined.

The use of traditional PD control is very popular not only because of its simple structure and easy implementation but also its acceptable performance for industrial applications. It is known that the PD control can be used for trajectory tracking with the asymptotic stability if the

* Corresponding author. Tel.: +1 306 966 5478; fax: +1 306 966 5427.
E-mail addresses: chris.zhang@usask.ca, chris_zhang@engr.usask.ca (W.J. Zhang).

0957-4158/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.mechatronics.2005.08.002

control gains are carefully selected [2–4]. However, the PD control is not satisfactory for applications which require high tracking accuracy. This limitation with the PD control is simply due to the inherent ''mismatch'' between the nonlinear dynamics behaviour of a manipulator and the linear regulating behaviour of the PD controller. Such a limitation is also true for the PID control.

The adaptive control can cope with parameter uncertainties, such as the link length, mass, inertia, and frictional nonlinearity, with a self-organizing capability. Having such a capability, however, requires extensive computation and thus compromises its application for real-time control problems (especially in high-speed operations). In addition, since the adaptive control generally does not guarantee that the estimated parameters of the manipulators converge to their true values [18], the tracking errors would repeatedly be brought into the system as the manipulators repeat their tasks.

Robot manipulators are usually used for repetitive tasks. In this case, the reference trajectory is repeated over a given operation time. This repetitive nature makes it possible to apply ILC to improve the tracking performance from iteration to iteration. It should be noted that ILC can be further classified into two kinds: off-line learning and on-line learning. In the case of off-line learning control, information in the controlled torque in the current iteration does not come from the current iteration but from the previous one. Philosophically, the learning in this case is shifted to the off-line mode. This then releases a part of the control workload at real-time, which implies the improvement of real-time trajectory tracking performance. In the case of the on-line learning control, the feedback control decision incorporates ILC at real-time.

Another active area of research in the control theory is the switching control [19–22]. In the switching control technique, the control of a given plant can be switched among several controllers, and each controller is designed for a specific ''nominal model'' of the plant. A switching control scheme usually consists of an inner loop (where a candidate controller is connected in closed-loop with the system) and an outer loop (where a supervisor decides which controller to be used and when to switch to a different one). As such, the switching of controllers is taken place in the time domain. This underlying philosophy may be modified to perform such a switching with respect to the iteration of learning.

In this paper, we present a new control method. The basic concept of this new control method is to combine several control methods by taking advantage of each of them into a hybrid one. The architecture of this hybrid control method is as follows: (1) the control is a learning process through several iterations of off-line operations of a manipulator, (2) the control structure consists of two parts: a PD feedback part and a feedforward learning part using the torque profile obtained from the previous iteration, and (3) the gains in the PD feedback law are adapted according to the gain switching strategy with respect to the iteration. This new control method is called the adaptive switching learning PD (ASL-PD) control method.

The remainder of the paper is organized as follows. In Section 2, the ASL-PD control method is described, and its features are discussed. Section 3 is devoted to the analysis of the asymptotic convergence of the ASL-PD control method using the Lyapunov's method. In Section 4, simulation studies are presented in which the ASL-PD method is compared with others. Conclusions are given in Section 5.

2. Adaptive switching learning PD control scheme

2.1. Dynamic model of a robot manipulator

Consider a robot manipulator with n joints running in repetitive operations. Its dynamics can be described by a set of nonlinear differential equations in the following form [1]:

$$D(q^j(t))\ddot{q}^j(t) + C(q^j(t),\dot{q}^j(t))\dot{q}^j(t) + G(q^j(t),\dot{q}^j(t)) + T_a(t) = T^j(t) \tag{1}$$

where $t \in [0, t_f]$ denotes the time and $j \in \mathbb{N}$ denotes the operation or iteration number. $q^j(t) \in \mathfrak{R}^n$, $\dot{q}^j(t) \in \mathfrak{R}^n$, and $\ddot{q}^j(t) \in \mathfrak{R}^n$ are the joint position, joint velocity, and joint acceleration vectors, respectively. $D(q^j(t)) \in \mathfrak{R}^{n\times n}$ is the inertia matrix, $C(q^j(t),\dot{q}^j(t))\dot{q}^j(t) \in \mathfrak{R}^n$ denotes the vector containing the Coriolis and centrifugal terms, $G(q^j(t),\dot{q}^j(t)) \in \mathfrak{R}^n$ is the gravitational plus frictional force, $T_a(t) \in \mathfrak{R}^n$ is the repetitive unknown disturbance, and $T^j(t) \in \mathfrak{R}^n$ is the input torque vector.

It is common knowledge that robot manipulators have the following properties [1]:

(P1) $D(q^j(t))$ is a symmetric, bounded, and positive definite matrix;

(P2) The matrix $\dot{D}(q^j(t)) - 2C(q^j(t),\dot{q}^j(t))$ is skew symmetric. Therefore,

$$x^{\mathrm{T}}(\dot{D}(q^j(t)) - 2C(q^j(t),\dot{q}^j(t)))x = 0 \quad \forall x \in \mathfrak{R}^n$$

Assume that all parameters of the robot are unknown and that:

(A1) The desired trajectory $q_d(t)$ is of the third-order continuity for $t \in [0, t_f]$.

(A2) For each iteration, the same initial conditions are satisfied, which are

$$q_d(0) - q^j(0) = 0, \quad \dot{q}_d(0) - \dot{q}^j(0) = 0, \qquad \forall j \in \mathbb{N}.$$

2.2. ASL-PD controller design

The ASL-PD control method has two operational modes: the single operational mode and the iterative operational mode. In the single operational mode, the PD control feedback with the gain switching is used, where information from the present operation is utilized. In the iterative operational mode, a simple iterative learning control is applied as feedforward where information from previous operations is used. Together with these two operational modes, all information from the current and previous operations is utilized. Specially, the ASL-PD control method can be described as follows.

Consider the jth iterative operation for system (1) with properties (P1 and P2) and assumptions (A1 and A2) under the following control law:

$$T^j(t) = \underbrace{K_p^j e^j(t) + K_d^j \dot{e}^j(t)}_{\text{feedback}} + \underbrace{T^{j-1}(t)}_{\text{feedforward}} \qquad j = 0, 1, \ldots, N \tag{2}$$

with the following gain switching rule

$$\begin{cases} K_p^j = \beta(j) K_p^0 \\ K_d^j = \beta(j) K_d^0 \qquad j = 1, 2, \ldots, N \\ \beta(j+1) > \beta(j) \end{cases} \tag{3}$$

where $T^{-1}(t) = 0$, $e^j(t) = q_d(t) - q^j(t)$, $\dot{e}^j(t) = \dot{q}_d(t) - \dot{q}^j(t)$, and K_p^0 and K_d^0 are the initial PD control gain matrices that are diagonal positive definite. The matrices K_p^0 and K_d^0 are called the initial proportional and derivative control gains, while matrices K_p^j and K_d^j are the control gains of the jth iteration. $\beta(j)$ is the gain switching factor where $\beta(j) > 1$ for $j = 1, 2, \ldots, N$, and it is a function of the iteration number.

The gain switching law in (3) is used to adjust the PD gains from iteration to iteration. Such a switching in the ASL-PD control method acts not in the time domain but in the iteration domain. This is the main difference between the ASL-PD control method and the traditional switching control method (where switching occurs in the time domain). Therefore, the transient process of the switched system, which must be carefully treated in the case of the traditional switching control method, does not occur in the ASL-PD control method.

From (2) and (3) it can be seen that the ASL-PD control law is a combination of feedback (with the switching gain in each iteration) and feedforward (with the learning scheme). The ASL-PD control method possesses an adaptive ability, which is demonstrated by the adoption of different control gains in different iterations; see (3). Such a switching takes place at the beginning of each iteration. Therefore, a rapid convergence speed for the trajectory tracking can be expected.

Furthermore, in the ASL-PD control law, the learning occurs due to the memorization of the torque profiles generated by the previous iterations that include information about the dynamics of a controlled system. It should be noted that such learning is direct in the sense that it generates the controlled torque profile directly from the existing torque profile in the previous iteration without any modification.

Because of the introduction of the learning strategy in the iteration, the state of the controlled object changes from iteration to iteration. This requires an adaptive control to deal with those changes, and the ASL-PD has such an adaptive capability.

In the next section, the proof of the asymptotic convergence of the ASL-PD control method for both position tracking and velocity tracking will be given.

3. Asymptotic convergence with the ASL-PD method

Eq. (1) can be linearized along the desired trajectory $(q_d(t), \dot{q}_d(t), \ddot{q}_d(t))$ in the following way:

$$D(t)\ddot{e}^j(t) + [C(t) + C_1(t)]\dot{e}^j(t) + F(t)e^j(t) + n(\ddot{e}^j, \dot{e}^j, e^j, t) - T_a(t) = H(t) - T^j(t) \tag{4}$$

where $D(t) = D(q_d(t))$

$$C(t) = C(q_d(t), \dot{q}_d(t))$$

$$C_1(t) = \left.\frac{\partial C}{\partial \dot{q}}\right|_{q_d(t),\dot{q}_d(t)} \dot{q}_d(t) + \left.\frac{\partial G}{\partial \dot{q}}\right|_{q_d(t),\dot{q}_d(t)}$$

$$F(t) = \left.\frac{\partial D}{\partial q}\right|_{q_d(t)} \ddot{q}_d(t) + \left.\frac{\partial C}{\partial q}\right|_{q_d(t),\dot{q}_d(t)} \dot{q}_d(t) + \left.\frac{\partial G}{\partial q}\right|_{q_d(t)}$$

$$H(t) = D(q_d(t))\ddot{q}_d(t) + C(q_d(t), \dot{q}_d(t))\dot{q}_d(t) + G(q_d(t))$$

The term $n(\ddot{e}^j, \dot{e}^j, e^j, t)$ contains the higher order terms $\ddot{e}^j(t)$, $\dot{e}^j(t)$, and $e^j(t)$, and it can be negligible. Therefore, for the jth and $j+1$th iterations, Eq. (4) can be rewritten, respectively, as follows:

$$D(t)\ddot{e}^j(t) + [C(t) + C_1(t)]\dot{e}^j(t) + F(t)e^j(t) - T_a(t) = H(t) - T^j(t) \tag{5}$$

$$D(t)\ddot{e}^{j+1}(t) + [C(t) + C_1(t)]\dot{e}^{j+1}(t) + F(t)e^{j+1}(t) - T_a(t) = H(t) - T^{j+1}(t) \tag{6}$$

For the simplicity of analysis, let $K_p^0 = \Lambda K_d^0$ for the initial iteration, and define the following parameter:

$$y^j(t) = \dot{e}^j(t) + \Lambda e^j(t) \tag{7}$$

The following theorem can be proved.

Theorem. *Suppose robot system* (1) *satisfies properties* (P1, P2) *and assumptions* (A1, A2). *Consider the robot manipulator performing repetitive tasks under the ASL-PD control method* (2) *with the gain switching rule* (3). *The following should hold for all* $t \in [0, t_f]$

$$q^j(t) \overset{j\to\infty}{\to} q_d(t)$$

$$\dot{q}^j(t) \overset{j\to\infty}{\to} \dot{q}_d(t)$$

provided that the control gains are selected so that the following relationships hold:

$$l_p = \lambda_{\min}(K_d^0 + 2C_1 - 2\Lambda D) > 0 \tag{8}$$

$$l_r = \lambda_{\min}(K_d^0 + 2C + 2F/\Lambda - 2\dot{C}_1/\Lambda) > 0 \tag{9}$$

$$l_p l_r \geqslant \|F/\Lambda - (C + C_1 - \Lambda D)\|_{\max}^2 \tag{10}$$

where $\lambda_{\min}(A)$ *is the minimum eigenvalue of matrix* A, *and* $\|M\|_{\max} = \max\|M(t)\|$ *for* $0 \leqslant t \leqslant t_f$. *Here*, $\|M\|$ *represents the Euclidean norm of* M.

Proof. Define a Lyapunov function candidate as

$$V^j = \int_0^t \mathrm{e}^{-\rho\tau} y^{j^\mathrm{T}} K_d^0 y^j \,\mathrm{d}\tau \geqslant 0 \tag{11}$$

where $K_d^0 > 0$ is the initial derivative gain of PD control, and ρ is a positive constant.

Also, define $\delta y^j = y^{j+1} - y^j$ and $\delta e^j = e^{j+1} - e^j$. Then, from (7)

$$\delta y^j = \delta\dot{e}^j + \Lambda\delta e^j \tag{12}$$

and from (2)

$$T^{j+1}(t) = K_p^{j+1} e^{j+1}(t) + K_d^{j+1}\dot{e}^{j+1}(t) + T^j(t) \tag{13}$$

From (5)–(7), (12), (13), one can obtain the following equation:

$$D\delta\dot{y}^j + (C + C_1 - \Lambda D + K_d^{j+1})\delta y^j + (F - \Lambda(C + C_1 - \Lambda D))\delta e^j = -K_d^{j+1} y^j \tag{14}$$

From the definition of V^j, for the $j+1$th iteration, one can get

$$V^{j+1} = \int_0^t \mathrm{e}^{-\rho\tau} y^{j+1^\mathrm{T}} K_d^0 y^{j+1}\,\mathrm{d}\tau$$

Define $\Delta V^j = V^{j+1} - V^j$. Then from (11), (12) and (14), we obtain

$$\begin{aligned}
\Delta V^j &= \int_0^t \mathrm{e}^{-\rho\tau}(\delta y^{j^\mathrm{T}} K_d^0 \delta y^j + 2\delta y^{j^\mathrm{T}} K_d^0 y^j)\,\mathrm{d}\tau \\
&= \frac{1}{\beta(j+1)} \int_0^t \mathrm{e}^{-\rho\tau}(\delta y^{j^\mathrm{T}} K_d^{j+1} \delta y^j + 2\delta y^{j^\mathrm{T}} K_d^{j+1} y^j)\,\mathrm{d}\tau \\
&= \frac{1}{\beta(j+1)} \Big\{ \int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} K_d^{j+1}\delta y^j\,\mathrm{d}\tau - 2\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta\dot{y}^j\,\mathrm{d}\tau \\
&\quad - 2\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}}((C + C_1 - \Lambda D + K_d^{j+1})\delta y^j \\
&\quad + (F - \Lambda(C + C_1 - \Lambda D))\delta y^j)\,\mathrm{d}\tau \Big\}
\end{aligned}$$

Applying the partial integration and from (A2), we have

$$\begin{aligned}
&\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta\dot{y}^j\,\mathrm{d}\tau \\
&= \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta y^j\Big|_0^t - \int_0^t (\mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D)'\delta y^j\,\mathrm{d}\tau \\
&= \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}}(t) D(t)\delta y^j(t) + \rho\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta y^j\,\mathrm{d}\tau \\
&\quad - \int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta\dot{y}^j\,\mathrm{d}\tau - \int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}}\dot{D}\delta y^j\,\mathrm{d}\tau
\end{aligned}$$

From (P1), one can get

$$\int_0^t \delta y^{j^\mathrm{T}}\dot{D}\delta y^j\,\mathrm{d}\tau = 2\int_0^t \delta y^{j^\mathrm{T}} C\delta y^j\,\mathrm{d}\tau$$

Then

$$\begin{aligned}
\Delta V^j = \frac{1}{\beta(j+1)} \Big\{ &-\mathrm{e}^{-\rho t}\delta y^{j^\mathrm{T}}(t) D(t)\delta y^j(t) - \rho\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta y^j\,\mathrm{d}\tau \\
&- 2\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}}(F - \Lambda(C + C_1 - \Lambda D))\delta e^j\,\mathrm{d}\tau \\
&- \int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}}(K_d^{j+1} + 2C_1 - 2\Lambda D)\delta y^j\,\mathrm{d}\tau \Big\}
\end{aligned} \tag{15}$$

From (3), we have

$$\begin{aligned}
\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} K_d^{j+1}\delta y^j\,\mathrm{d}\tau &= \beta(j+1)\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} K_d^0\delta y^j\,\mathrm{d}\tau \\
&\geqslant \int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} K_d^0\delta y^j\,\mathrm{d}\tau
\end{aligned} \tag{16}$$

Substituting (12) into (15) and noticing (16), we obtain

$$\begin{aligned}
\Delta V^j \leqslant \frac{1}{\beta(j+1)} \Big\{ &-\mathrm{e}^{-\rho t}\delta y^{j^\mathrm{T}}(t) D(t)\delta y^j(t) - \rho\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta y^j\,\mathrm{d}\tau \\
&- \int_0^t \mathrm{e}^{-\rho\tau}\delta\dot{e}^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta\dot{e}^j\,\mathrm{d}\tau \\
&- 2\Lambda\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta\dot{e}^j\,\mathrm{d}\tau \\
&- 2\int_0^t \mathrm{e}^{-\rho\tau}\delta\dot{e}^{j^\mathrm{T}}(F - \Lambda(C + C_1 - \Lambda D))\delta e^j\,\mathrm{d}\tau \\
&- \Lambda^2\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta e^j\,\mathrm{d}\tau \\
&- 2\Lambda\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(F - \Lambda(C + C_1 - \Lambda D))\delta e^j\,\mathrm{d}\tau \Big\}
\end{aligned}$$

Applying the partial integration again gives

$$\begin{aligned}
&\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta\dot{e}^j\,\mathrm{d}\tau \\
&= \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta e^j\Big|_0^t \\
&\quad + \rho\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta e^j\,\mathrm{d}\tau \\
&\quad - \int_0^t \mathrm{e}^{-\rho\tau}\delta\dot{e}^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta e^j\mathrm{d}\tau \\
&\quad + 2\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(\Lambda\dot{D} - \dot{C}_1)\delta e^j\,\mathrm{d}\tau
\end{aligned}$$

Therefore,

$$\begin{aligned}
\Delta V^j &\leqslant \frac{1}{\beta(j+1)} \Big\{ -\mathrm{e}^{-\rho t}\delta y^{j^\mathrm{T}} D\delta y^j - \rho\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta y^j\,\mathrm{d}\tau \\
&\quad - \Lambda e^{-\rho t}\delta e^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta e^j \\
&\quad - \rho\Lambda\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta e^j\,\mathrm{d}\tau - \int_0^t \mathrm{e}^{-\rho\tau} w\,\mathrm{d}\tau \Big\} \\
&\leqslant \frac{1}{\beta(j+1)} \Big\{ -\mathrm{e}^{-\rho t}\delta y^{j^\mathrm{T}} D\delta y^j - \Lambda\mathrm{e}^{-\rho t}\delta e^{j^\mathrm{T}} l_p\delta e^j \\
&\quad - \rho\int_0^t \mathrm{e}^{-\rho\tau}\delta y^{j^\mathrm{T}} D\delta y^j\,\mathrm{d}\tau - \rho\Lambda\int_0^t \mathrm{e}^{-\rho\tau}\delta e^{j^\mathrm{T}} l_p\delta e^j\,\mathrm{d}\tau \\
&\quad - \int_0^t \mathrm{e}^{-\rho\tau} w\,\mathrm{d}\tau \Big\}
\end{aligned} \tag{17}$$

where

$$w = \delta\dot{e}^{j\mathrm{T}}(K_d^0 + 2C_1 - 2\Lambda D)\delta\dot{e}^j + 2\Lambda\delta\dot{e}^{j\mathrm{T}}(F/\Lambda - (C + C_1 - \Lambda D))\delta e^j + \Lambda^2\delta e^{j\mathrm{T}}(K_d^0 + 2C + 2F/\Lambda - 2\dot{C}_1/\Lambda)\delta e^j$$

Let $Q = F/\Lambda - (C + C_1 - \Lambda D)$. Then from (8) and (9), we obtain

$$w \geqslant l_p\|\delta\dot{e}\|^2 + 2\Lambda\delta\dot{e}^{\mathrm{T}}Q\delta e + \Lambda^2 l_r\|\delta e\|^2$$

Applying the Cauchy–Schwartz inequality gives

$$\delta\dot{e}^{\mathrm{T}}Q\delta e \geqslant -\|\delta\dot{e}\|\,\|Q\|_{\max}\|\delta e\|$$

From (8)–(10)

$$w \geqslant l_p\|\delta\dot{e}\|^2 - 2\Lambda\|\delta\dot{e}\|\,\|Q\|_{\max}\|\delta e\| + \Lambda^2 l_r\|\delta e\|^2 = l_p\left(\|\delta\dot{e}\| - \frac{\Lambda}{l_p}\|Q\|_{\max}\|\delta e\|\right)^2 + \Lambda^2\left(l_p - \frac{1}{l_r}\|Q\|_{\max}^2\right)\|\delta e\|^2 \geqslant 0 \tag{18}$$

From (P1) and (8), based on (17), it can be ensured that $\Delta V^j \leqslant 0$. Therefore,

$$V^{j+1} \leqslant V^j \tag{19}$$

From the definition, K_d^0 is a positive definite matrix. From the definition of V^j, $V^j > 0$, and V^j is bounded. As a result, $y^j(t) \to 0$ when $j \to \infty$. Because $e^j(t)$ and $\dot{e}^j(t)$ are two independent variables, and Λ is a positive constant. Thus, if $j \to \infty$, then $e^j(t) \to 0$ and $\dot{e}^j(t) \to 0$ for $t \in [0, t_f]$.

Finally, the following conclusions hold

$$\begin{cases} q^j(t) \overset{j\to\infty}{\to} q_d(t) \\ \dot{q}^j(t) \overset{j\to\infty}{\to} \dot{q}_d(t) \end{cases} \quad \text{for } t \in [0, t_f] \tag{20}$$

From the above analysis it can be seen that the ASL-PD control method can guarantee that the tracking errors converge arbitrarily close to zero as the number of iterations increases. The following case studies based on simulation will demonstrate this conclusion. □

4. Simulation

In order to have some idea about how effective the proposed ASL-PD control method would be, we conducted a simulation study; specifically we simulated two robot manipulators. The first one was a serial robot manipulator with parameters taken directly from [6] for the purpose of comparing the ASL-PD method with the method proposed in Ref. [6] called the adaptive ILC. It is noted that the result for the serial manipulator may not be applicable to the parallel manipulator. Therefore, the second one is a parallel robot manipulator for which we show the effectiveness of the ASL-PD control method both in the trajectory tracking error and the required torque in the motor.

4.1. Trajectory tracking of a serial robot manipulator

A two degrees of freedom (DOF) serial robot is shown in Fig. 1, which was discussed in [6] with an adaptive ILC method.

The physical parameters and desired trajectories are the same as in [6] and listed as follows.

Physical parameters:

$m_1 = 10$ kg, $m_2 = 5$ kg, $l_1 = 1$ m, $l_2 = 0.5$ m, $l_{c_1} = 0.5$ m, $l_{c_2} = 0.25$ m, $I_1 = 0.83$ kg m^2 and $I_2 = 0.3$ kg m^2.

Desired trajectories and the repetitive disturbances:

$$q_1 = \sin 3t, \quad q_2 = \cos 3t \quad \text{for } t \in [0, 5]$$
$$d_1(t) = a0.3\sin t, \quad d_2(t) = a0.1(1 - \mathrm{e}^{-t}) \quad \text{for } t \in [0, 5]$$

where a is a constant used to examine the capability of the ASL-PD control to deal with the repetitive disturbances.

The control gains were also set to be the same as [6]

$$K_p^0 = K_d^0 = \mathrm{diag}\{20, 10\}$$

In the ASL-PD control method, the control gains were switched from iteration to iteration based on the following rule:

$$K_p^j = 2jK_p^0, \quad K_d^j = 2jK_d^0, \qquad j = 1, 2, \ldots, N$$

First, consider $a = 1$. In that way, the repetitive disturbances were the same as [6]. Fig. 2a shows the tracking performance for the initial iteration, where only the PD control with small control gains was used, and no feedforward is used. It can be seen that the tracking performance was not acceptable because the errors were too large for both joints. However, at the sixth iteration where the ASL-PD control method was applied, the tracking performance was improved dramatically as shown in Fig. 2b. At the eighth iteration, the performance was very good (Fig. 2c).

The velocity tracking performance is shown in Fig. 3. From it one can see that the velocity errors reduced from 1.96 (rad/s) at the initial iteration to 0.0657 (rad/s) at the sixth iteration, and further to 0.0385 (rad/s) at the eighth iteration for joint 1. The similar decreasing trend can be found for joint 2. From Figs. 2 and 3 it can be seen that

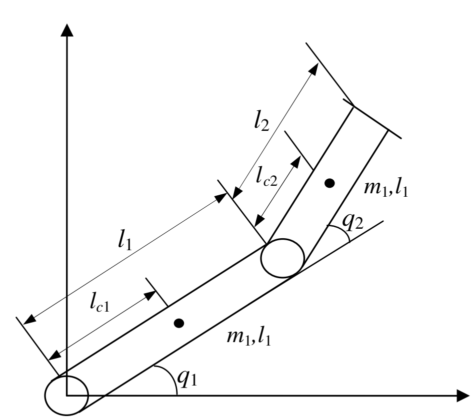

Fig. 1. Configuration of a serial robot manipulator.

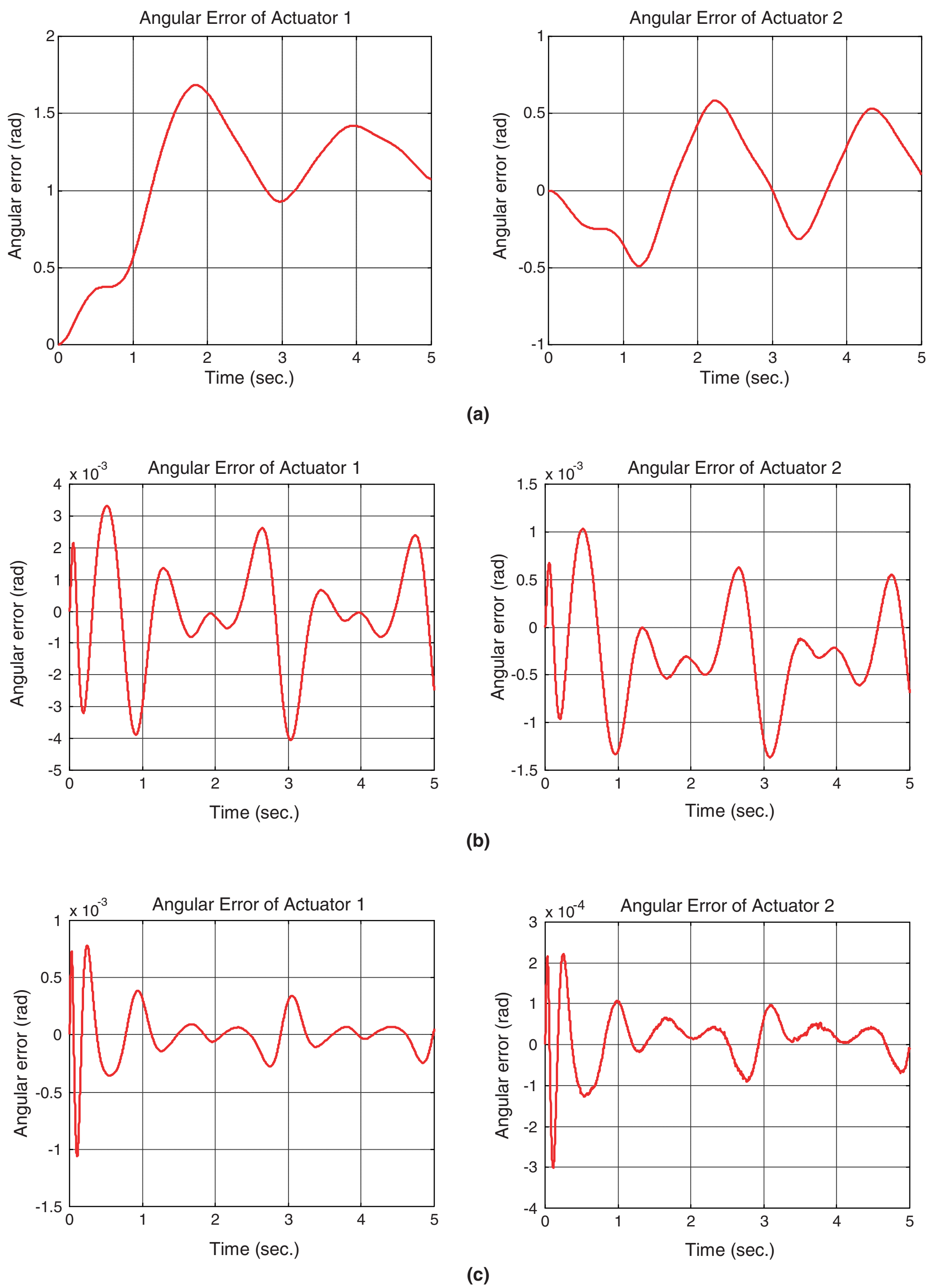

Fig. 2. Position tracking errors for different iterations under ALS-PD control. (a) Angular errors for two joints in the initial iteration, (b) angular errors for two joints at the sixth iteration and (c) angular errors for two joints at the eighth iteration.

the tracking performances were improved incrementally with the increase of the iteration number.

As the gain switching rule was introduced at each iteration, the convergence rate increased greatly compared with the control method developed in [6]. Table 1 shows the trajectory tracking errors from the initial iteration to the eighth iteration. From Table 1 it can be seen that the tracking performance was considerably improved at the sixth iteration. The maximum position errors for joints 1 and 2 were 0.0041 rads and 0.0014 rads, respectively, while the similar results were achieved after 30 iterations using the adaptive ILC in [6]. (The maximum position errors for

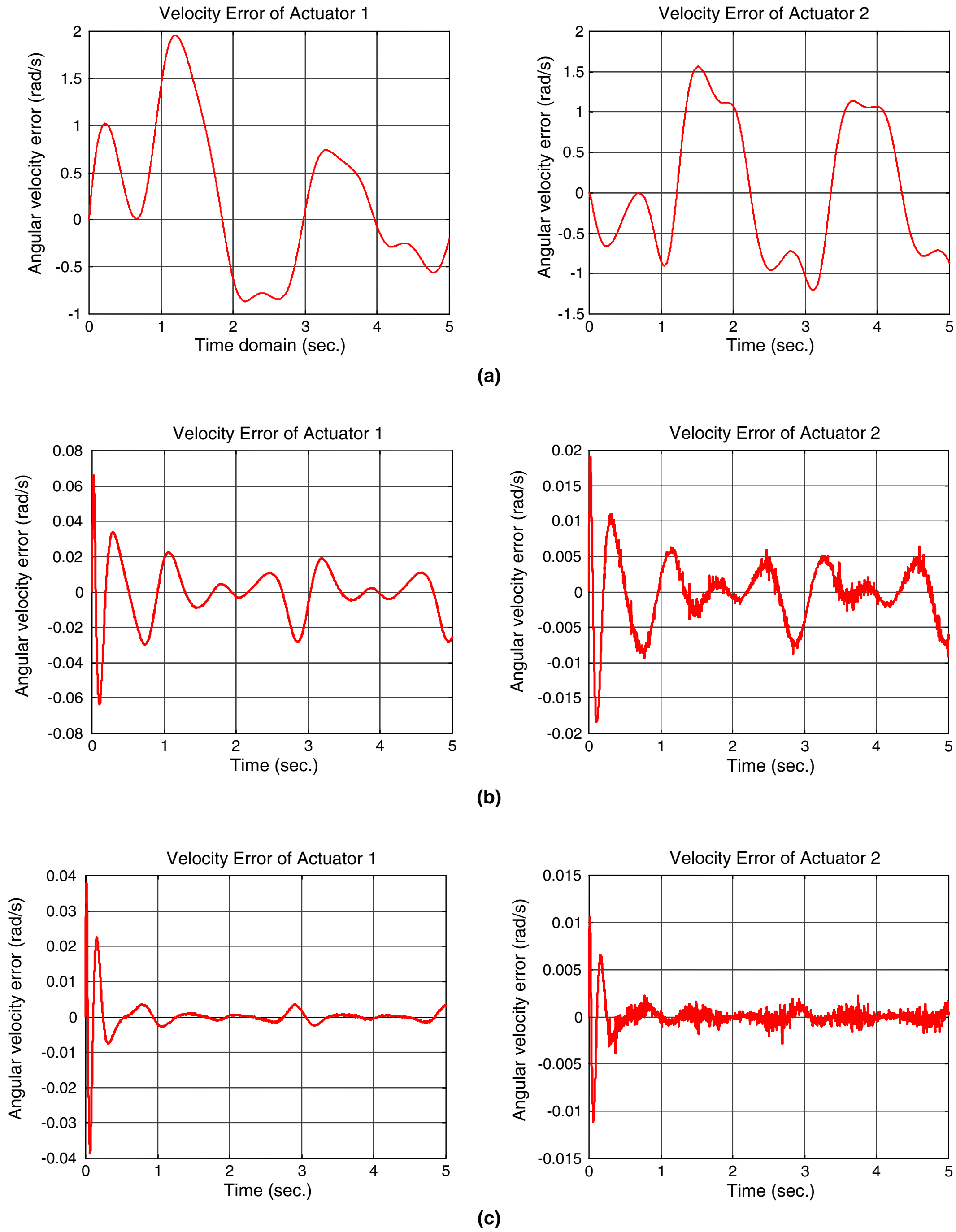

Fig. 3. Velocity tracking errors for different iterations under ASL-PD control. (a) Velocity errors for two joints in the initial iteration, (b) velocity errors for two joints at the sixth iteration and (c) velocity errors for two joints at the eighth iteration.

Table 1
Trajectory tracking errors from iteration to iteration

	Iteration				
	0	2	4	6	8
max $\lvert e_1^j \rvert$(rad)	1.6837	0.4493	0.0433	0.0041	0.0011
max $\lvert e_2^j \rvert$(rad)	0.5833	0.1075	0.0122	0.0014	3.01E−4
max $\lvert \dot{e}_1^j \rvert$(rad/s)	1.9596	0.7835	0.1902	0.0657	0.0385
max $\lvert \dot{e}_2^j \rvert$(rad/s)	1.5646	0.2534	0.0523	0.0191	0.0111

joints 1 and 2 were 0.0041 and 0.0046 (rad), respectively.) Therefore, the comparison of their method and our method demonstrates a fast convergence rate with the ASL-PD control method. It should be noted that the comparison of the velocity errors was not done as such information was not presented in Ref. [6].

It is further noted that there were repetitive disturbances at each iteration in the simulation. To examine the capacity of the ASL-PD under the repetitive disturbance condition, different levels of the repetitive disturbances were applied in

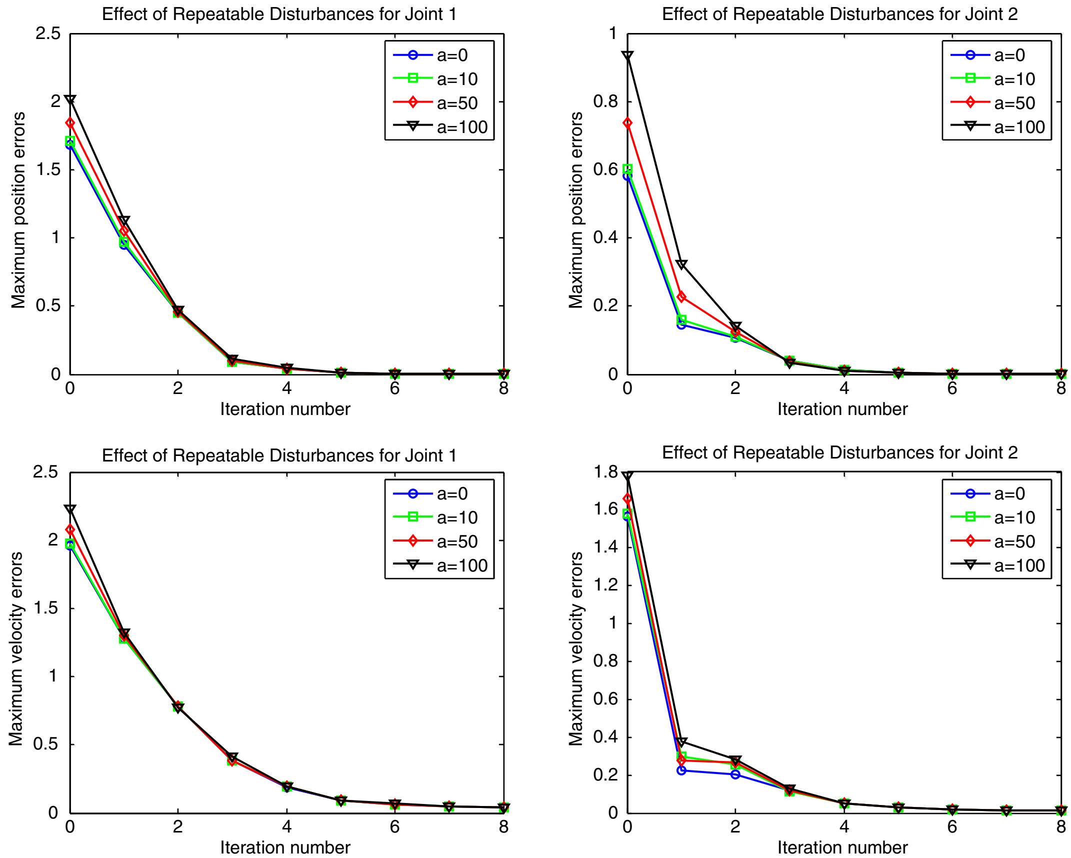

Fig. 4. Effect of the repetitive disturbance on tracking errors.

the simulation. Fig. 4 shows the maximum tracking errors from iteration to iteration for different repetitive disturbances which are expressed by a constant a. A larger constant a means a more disturbance. It should be noted that in Fig. 4 $a = 0$ means there is no repetitive disturbance in the simulation, and $a = 100$ means a large repetitive disturbance included in the simulation (specifically, the disturbance acted in joint 1 was about 20% of the required torque and the disturbance acted in joint 2 was about 40% of the required torque). From this figure, one can see that although the tracking errors for the initial iteration increased with the increase of the disturbance level, the final tracking errors of both the position and the velocity were the same for the different repetitive disturbance levels at the final two iterations. Therefore, we conclude that the ASL-PD control method has an excellent capability in terms of both rejecting the repetitive disturbance and robustness with respect to the disturbance level.

4.2. Trajectory tracking of a parallel robot manipulator

A two DOFs parallel robot manipulator is shown in Fig. 5. Table 2 lists its physical parameters. The robot system can be viewed as two serial robotic systems with some constraints; that is, the two end-effectors of these two serial robotic systems reach the same position. Because of this constraint, the dynamics is more complex than that of its serial counterpart. The details about the dynamics of the parallel robot manipulator can be founded in [23].

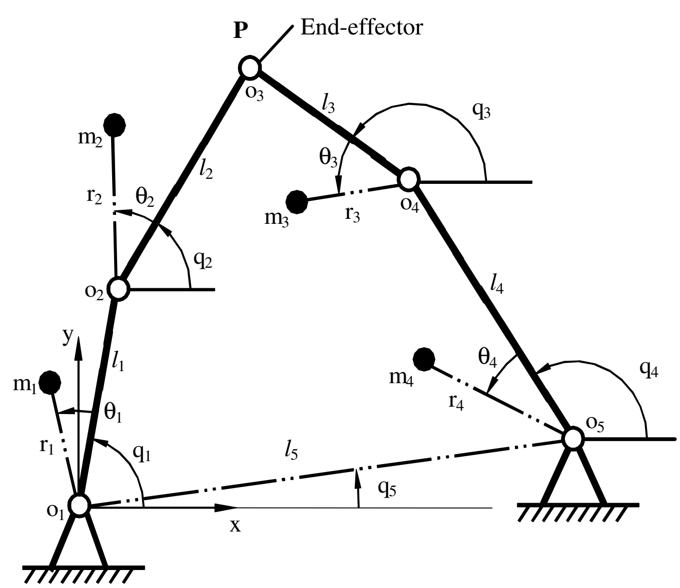

Fig. 5. Scheme of a two DOFs parallel robot manipulator.

The end-effector of the robot was required to move from point A (0.7, 0.3), to point B (0.6, 0.4), and to point C (0.5, 0.5). The time duration between two nearby points

Table 2
Physical parameters of the parallel robotic manipulator

Link	m_i (kg)	l_i (m)	r_i (m)	I_i (kg m^2)	θ_i (rad)
1	1	0.4	0.2	0.5	0
2	1.25	0.6	0.3	1	0
3	1.5	0.8	0.3	1	0
4	1	0.6	0.2	0.5	0
5	–	0.6	–	–	–

was 0.25 s. The control was carried out at the joint level where the inverse kinematics was used to calculate the joint position and velocity associated with the specific path of the end-effector. The path was designed to pass through these three points with the objective of meeting the positions, velocities, and accelerations at these three points using the motion planning method [24].

In this example, the control gains were selected as follows:

$$K_p^0 = \mathrm{diag}\{20, 20\}, \quad K_d^0 = \mathrm{diag}\{12, 12\}$$

The gain switching rule was set to be

$$K_p^j = 2jK_p^0, \qquad K_d^j = 2jK_d^0 \qquad \text{for } j = 1, 2, \ldots, N$$

Fig. 6 shows the position tracking performance improvement for the two actuators from iteration to iteration. From it one can see that, at the initial iteration, the maximum position errors were about 0.11 and 0.38 rad; only after four iterations, the maximum position errors were reduced to 0.08 and 0.05 rad; finally, after eight iterations, the maximum errors were reduced to 0.0003 and 0.0008 rad. Fig. 7 shows the velocity tracking performance improvement for the two actuators. At the initial iteration, the maximum velocity errors were about 1.17 and 2.68 rad/s in the two actuators, respectively. But after four iterations, the maximum values were reduced to 0.15 and 0.14 rad/s. After eight iterations, the maximum errors in the two actuators became 0.0046 and 0.0102 rad/s for velocity, respectively.

It should be noted that, while the tracking performance was improved from iteration to iteration, the torques required to drive the two actuators were nearly the same from iteration to iteration after a few iterations. This can be seen from Fig. 8, especially from the fifth iteration to the eighth iteration. It can be seen also from Fig. 8 that the profiles of the required torques were very smooth even as the control gains become larger as the iteration number is increased. Such a property is very useful for the safe use of the actuators and the attenuation of vibration of the controlled plant. It is noted that this property was missed in the switching technique in the time domain.

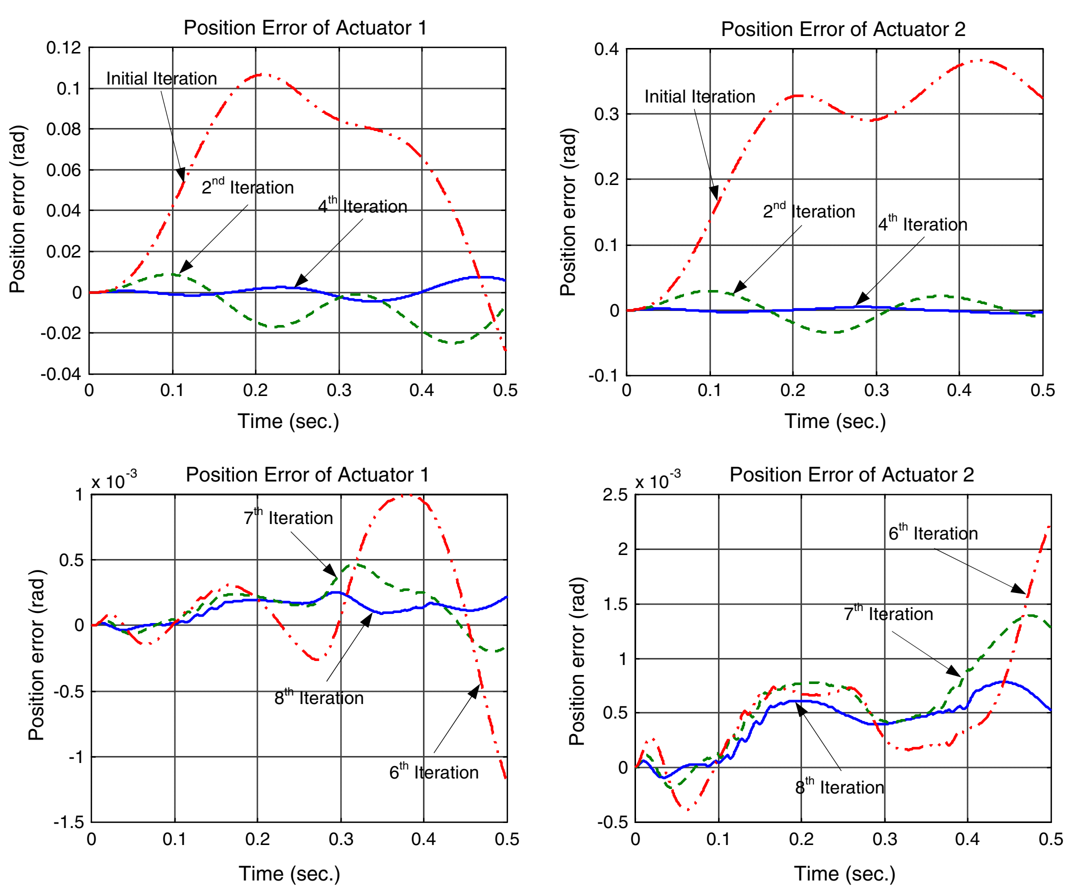

Fig. 6. Position tracking performance improvement from iteration to iteration.

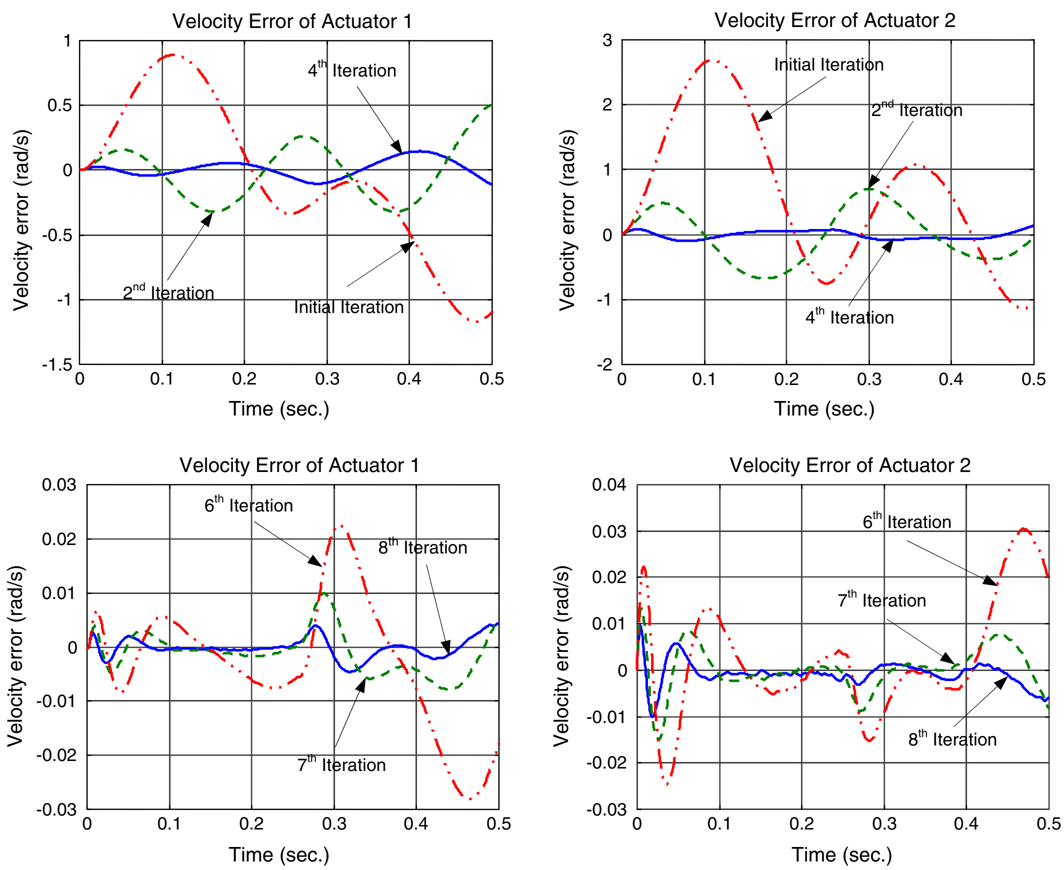

Fig. 7. Velocity tracking performance improvement from iteration to iteration.

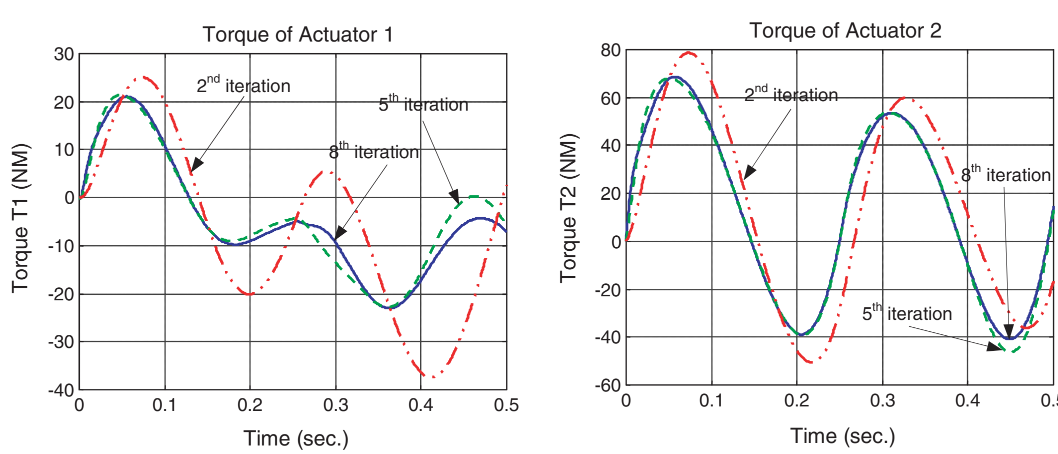

Fig. 8. The required torque profiles for iteration $j = 2, 5, 8$.

5. Conclusion

In this paper, a new adaptive switching learning PD (ASL-PD) control method is proposed. This control method is a simple combination of a traditional PD control with a gain switching strategy as feedback and an iterative learning control using the input torque profile obtained from the previous iteration as feedforward. The ASL-PD control incorporates both adaptive and learning capabilities; therefore, it can provide an incrementally improved tracking performance with the increase of the iteration number. The ASL-PD control method achieves the asymptotic convergence based on the Lyapunov's method. The position and velocity tracking errors monotonically decrease with the increase of the iteration number. The concept of integrating the switching technique and the iterative learning scheme works very well; especially with the achievement of a fast convergence speed. The simulation study has demonstrated the effectiveness of the ASL-PD control method. Its distinct features are the simple structure, easy implementation, fast convergence, and excellent performance.

Acknowledgement

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) through a PGS-B scholarship to the first author and partially supported by NSERC through a strategy project research grant to the second and third authors.

References

[1] Craig JJ. Introduction to robotics: mechanics and control. Reading, MA: Addison-Wesley; 1986.
[2] Qu ZH. Global stability of trajectory tracking of robot under PD control. Dynam Contr 1995;5(1):59–71.
[3] Kerry R. PD control with desired gravity compensation of robotic manipulators: a review. Int J Robot Res 1997;16(5):660–72.
[4] Chen QJ, Chen HT, Wang YJ, Woo PY. Global stability analysis for some trajectory tracking control schemes of robotic manipulators. J Robot Syst 2001;18(2):69–75.
[5] Craig JJ. Adaptive control of mechanical manipulators. Addison-Wesley; 1988.
[6] Choi JY, Lee JS. Adaptive iterative learning control of uncertain robotic systems. IEE Proc Contr Theory Appl 2000;147(2):217–23.
[7] Slotine JJ, Li W. On the adaptive control of robot manipulators. Int J Robot Res 1987;6(3):49–59.
[8] Li Q, Poo AN, Teo CL, Lim CM. Developing a neuro-compensator for the adaptive control of robots. IEE Proc Cont Theory Appl 1996;142(6):562–8.
[9] Li Q, Poo AN, Teo CL. A multi-rate sampling structure for adaptive robot control using a neuro-compensator. Artificial Intell Eng 1996;10(1):85–94.
[10] Tomei P. Adaptive PD controller for robot manipulators. IEEE Trans Robot Automation 1991;7(4):565–70.
[11] Li Q, Tso SK, Zhang WJ. Trajectory tracking control of robot manipulators using a neural-network-based torque-compensator. Proc I Mech E, Part I, J Syst Contr 1998;212(5):361–72.
[12] Arimoto S, Kawamura S, Miyasaki F. Bettering operation of robots by learning. J Robot Syst 1984;1(2):123–40.
[13] Kawamura S, Miyazaki F, Arimoto S. Realization of robot motion based on a learning method. IEEE Trans Syst Man Cybernet 1988;18(1):123–6.
[14] Tayebi A. Adaptive iterative learning control for robot manipulators. In: Proceedings of the American control conference, Denver, CO, USA, June 2003. p. 4518–23.
[15] Kuc TY, Nam K, Lee JS. An iterative learning control of robot manipulators. IEEE Trans Robot Automat 1991;7(6):835–42.
[16] Chen YQ, Moore KL. PI-type iterative learning control revisited. In: Proceedings of the American control conference, Anchorage, AK, USA, May 2002. p. 2138–43.
[17] Yan XG, Chen IM, Lam J. D-type learning control for nonlinear time-varying systems with unknown initial states and inputs. Trans Inst Measure Contr 2001;23(2):69–82.
[18] Sun D, Mills JK. Performance improvement of industrial robot trajectory tracking using adaptive-learning scheme. Trans ASME, J Dynam Syst Measure Contr 1999;121(2):285–92.
[19] Nussbaum RD. Some remarks on a conjecture in parameter adaptive control. Syst Contr Lett 1983;3:243–6.
[20] Middleton RH, Goodwin GC, Hill DJ, Mayne DQ. Design issues in adaptive control. IEEE Trans Automat Contr 1988;33(1):50–8.
[21] Fu M, Barmish BR. Adaptive stabilization of linear systems via switching control. IEEE Trans Automat Contr 1986;31(6): 1097–103.
[22] Martensson B. The order of any stabilizing regulator is sufficient a priori information for adaptive stabilizing. Syst Contr Lett 1985;6(2):87–91.
[23] Ouyang PR, Li Q, Zhang WJ, Guo LS. Design, modeling and control of a hybrid machine system. Mechatronics 2004;14(10):1197–217.
[24] Ouyang PR, Zhang WJ. Force balancing of robotic mechanisms based on adjustment of kinematic parameters. ASME J Mech Des 2005;127(3):433–40.