Proceedings of the 1992 IEEE
International Conference on Robotics and Automation
Nice, France - May 1992

Iterative Learning Control of Robots with Elastic Joints

Alessandro De Luca   Giovanni Ulivi

Dipartimento di Informatica e Sistemistica
Università degli Studi di Roma "La Sapienza"
Via Eudossiana 18, 00184 Roma, Italy

Abstract

Learning control has been successfully applied for tracking repetitive trajectories in robot manipulators, under the assumption of perfect rigidity of links and transmissions. When joint elasticity is non-negligible, the usual convergence conditions fail to hold due to the lack of passivity between motor torque input and link velocity output. More recently, convergence conditions have been derived that guarantee learning convergence under milder assumptions, by giving up perfect tracking of reference signals beyond a certain frequency. The case of robots with joint elasticity is used here to illustrate the usefulness of this frequency-based design methodology. An efficient and simple iterative learning algorithm is presented, allowing to solve the output tracking problem with limited knowledge of system dynamics and using only a linear stabilizing feedback on the motor variables. Simulations reported for a two-link planar robot under gravity show good motion performance also in this critical case.

1. Introduction

Learning control of robot manipulators is a technique allowing the correct execution of repetitive tasks through iterative training [1,2]. In order to learn the motor torque required for trajectory reproduction, only a reduced knowledge of the robot dynamic model is used. A priori information is limited to basic properties owned by the mechanical system, such as passivity of the mapping from torque to link velocity [3], rather than accurate estimates of the model parameters. The learning process can also be interpreted as a simple (i.e. model independent) computational scheme for obtaining the so-called inverse dynamics solution to the tracking control problem, without introducing high-gain feedback.

The learning approach has been shown to be robust w.r.t. measurement noise [4], but other phenomena like actuator dynamics, link and transmission flexibility, or measurement delays are usually not included in the theoretical analysis. When some of these effects are not negligible, the standard convergence conditions no longer hold; eventually, closed-loop stability may be destroyed. In experiments, unmodeled dynamics concentrate their actions at high frequencies and the outcome is a degradation of the learning process in the long run. This drawback was noticed by several researchers [5,6].

While exact learning algorithms are quite sensitive to the above effects, a frequency-based approach has been introduced in [7] which efficiently handles the problem of unmodeled dynamics. The proposed learning algorithm is designed as a combined filter, generating feedforward term for the next trial by a frequency-dependent weighting of both the control effort (viz. the error) at the current trial and of the previous memory. Convergence conditions can be derived that are weaker than the usual ones, compromising with the accuracy of the learned behavior.

The design tools used are standard in the engineering practice: linearization around a nominal operating point, Nyquist diagrams, and linear filtering. The ease and flexibility of use of this learning method has been demonstrated also experimentally on several robotic systems, ranging from conventional manipulators [7], to redundant robots [8], and to one-link flexible arms [9,10]. The resulting control scheme is very simple and can be implemented with a limited computing power.

In this paper, we apply our learning method to the case of robots with rigid links but with elastic joints. Increasing interest is now being devoted to the control of this class of robot arms [11–13], due to their relevance in current industrial environments. In fact, elasticity concentrated at the joints accounts for the presence of common transmission elements like harmonic drives, belts, or long shafts, all of which are deformable.

Accurate trajectory control of robots with joint elasticity is recognized to be a difficult problem, even when the dynamic model is accurately known. This class of nonlinear systems does not fulfill in general the necessary conditions for exact linearization or decoupling via static state feedback —the basis for the computed torque method [14]. The resort to nonlinear dynamic state feedback solves the problem but requires a very complex control law [15]. Also, there is no passivity property between the motor torque input and the link velocity output [16]. This gives rise to problems also for adaptive schemes, which are only at a beginning stage [17,18], as well as for convergence of standard learning techniques.

In the absence of a reliable dynamic model, there are very few results on learning of repetitive trajectories in joint elastic robots, mainly reported in [19]. One of the major limitations relies in the non-collocation between the reference trajectory defined at the link level —which is one-to-one related to the end-effector position— and the desired input to be learned, the torque at the motor level. This motivated the 'two-stage betterment' approach of [19]: first, the motor reference trajectory is iteratively acquired; next, the required input torques are learned. These two intertwined processes slow down considerably the overall convergence. Moreover, the analysis is performed with the simplified dynamic model of [12], while numerical results were obtained only for slow trajectories and no gravity. Considerable problems were reported due to high-frequency noise.

Here, we face the learning problem directly in a 'one-stage' fashion. In particular, the following convenient facts will be used:

- A model independent feedback controller is included for a better shaping of the closed-loop behavior of the system. In particular, global stabilization can be accomplished by feeding back only motor variables [20].
- For robot arms with elastic joints, even described by a general dynamic model [20,21], invertibility of the plant is always guaranteed. In fact, an algebraic relationship can be derived between the desired trajectory (and its time derivatives) of the link outputs and the required input motor torques [22].
- Since the learning algorithm is to be applied off-line between one trial and the successive one, noncausal digital techniques can be used in the signal filtering in order to avoid learning oscillations and increase the speed of convergence.

The paper is organized as follows. In Section 2, relevant properties of the dynamic model of robots with joint elasticity are presented. Next, the general learning methodology is briefly recalled. In Section 4, the actual steps in the design of the learning filters are outlined for a two-link planar arm with rotational joints moving under gravity. Simulation results on fast joint trajectories are reported in Section 5.

2. Properties of the Dynamic Model

The general dynamic model of a robot arm with N elastic joints has the block-partitioned form [20,21]

$$\begin{bmatrix} M_1(q) & M_2(q) \\ M_2^T(q) & M_3 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} K_E & -K_E \\ -K_E & K_E \end{bmatrix} \begin{bmatrix} q \\ \theta \end{bmatrix} + \begin{bmatrix} H_{A1}(q,\dot{\theta}) + H_{B1}(q,\dot{q}) & H_{B2}(q,\dot{q}) \\ H_{B3}(q,\dot{q}) & 0 \end{bmatrix} \begin{bmatrix} \dot{q} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} F_l & 0 \\ 0 & F_m \end{bmatrix} \begin{bmatrix} \dot{q} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} e_1(q) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ u \end{bmatrix}, \quad (1)$$

where q and θ are N-vectors of link and motor coordinates respectively, and u is the actuator torque vector. In the $2N \times 2N$ positive definite inertia matrix, the block $M_2(q)$ is strictly upper triangular. Matrices H_{A1}, H_{B2}, and H_{B3} have a linear dependence on velocity, and are zero when M_2 is constant. Gravity terms are present only in e_1, since the center of mass of each motor is assumed to be on its own rotation axis. The positive coefficients of joint elasticity are contained in $K_E = \operatorname{diag}\{k_1, \ldots, k_N\}$, while F_m and F_l are diagonal positive semidefinite matrices, representing friction at the motor and at the link side of the transmissions.

In the following, we will restrict for simplicity our attention to:

(A) *M_2 constant.*

This is the case of a 2R planar arm or of a 3R manipulator with vertical trunk and no offsets. The resulting model displays a series of properties listed hereafter.

P1. *System equations (1) are in general neither linearizable nor input-output decouplable via static state feedback.*

P2. *The mapping between motor torque u and link velocity $\dot{q}$ is not passive.*

Instead, the mapping between motor torque u and velocity $\dot{\theta}$ still satisfies the passivity condition [18]. In spite of the above 'negative' results, the following properties are very helpful.

P3. *Fully actuated robots with joint elasticity have no zero dynamics* [15].

For this class of systems, a more general control law based on dynamic state feedback can be used to transform the closed-loop equations into fully linear and input-output decoupled ones.

P4. *Regulation at a constant link reference position can be achieved using only motor feedback plus a constant gravity compensation at the set-point* [20].

The stabilizing feedback at $q = q_d$ is

$$u = K_P(\theta_d - \theta) - K_D\dot{\theta} + e_1(q_d), \quad (2)$$

with $K_D > 0$ and sufficiently large $K_P > 0$, and where

$$\theta_d = q_d + K_E^{-1} e_1(q_d). \quad (3)$$

P5. *For any smooth reference trajectory $q_d(t)$ of the links, the associated trajectory $\theta_d(t)$ of the motor variables can be derived in closed-form* [22].

Under assumption **(A)**, this can be shown as follows. The first N equations evaluated at $q = q_d(t)$ yield

$$\begin{aligned} &M_1(q_d)\ddot{q}_d + M_2\ddot{\theta} + H_{B1}(q_d, \dot{q}_d)\dot{q}_d \\ &+ K_E(q_d - \theta) + F_l\dot{q}_d + e_1(q_d) = 0, \end{aligned} \quad (4)$$

that can be rewritten as

$$M_2\ddot{\theta} - K_E\theta = -w_d(t), \quad (5)$$

with a known forcing time function on the right-hand side. Due to the strictly upper triangular form of M_2, (5) is a *singular* linear differential equation, that can be solved recursively for θ as

$$\theta_{d,i}(t) = \frac{1}{k_i}\big(w_{d,i}(t) + \sum_{j=i+1}^{N}\{M_2\}_{ij}\ddot{\theta}_{d,j}(t)\big), \quad (6)$$

for $i = N-1, \ldots, 1$, with $\theta_{d,N}(t) = w_{d,N}(t)/k_N$. Accordingly, the nominal feedforward u_d required for the given output trajectory can be computed using (6) in the second set of N equations (1):

$$u_d(t) = M_2^{\mathrm{T}}\ddot{q}_d + M_3\ddot{\theta}_d + K_E(\theta_d - q_d) + F_m\dot{\theta}_d. \quad (7)$$

Equations (6) and (7) give an explicit form for the input to be acquired during the learning process and specify that the reference trajectory $q_d(t)$ must be piecewise C^3 for exact reproduction. They also imply that the required nominal input $u_d(t)$ is a *causal* one, i.e. it depends only on the current desired state of the robot arm.

3. Review of the Learning Method

In [7] a new frequency-based learning algorithm was introduced and its convergence was formally proven for linear systems.

Here, we follow the slightly different approach shown in Fig. 1. This scheme is more suitable for the case of robots with elastic joints, since it conveniently separates the stabilization phase from the learning one, needed for accurate tracking. In particular, we stress that the error used within the feedback control loop, can be different from the one used for updating the learning term.

For the ease of description, consider a linear scalar plant and let $y_d(t)$ be the desired repetitive time behavior for the system output y, and $e(t) = y_d(t) - y(t)$ the error. Assume that the plant has already been stabilized in a robust way, being $w(s) = y(s)/v(s)$ the resulting closed-loop transfer function. At the kth iteration ($k = 1, 2, \ldots$), the following relations hold among Laplace transforms of the involved signals:

$$\begin{aligned} v_k(s) &= y_d(s) + \mu_{k-1}(s) && \text{(reference input)} \\ y_k(s) &= w(s)v_k(s) && \text{(plant output)} \\ \mu_k(s) &= \alpha(s)e_k(s) + \beta(s)\mu_{k-1}(s) && \text{(learning law)} \end{aligned} \quad (8)$$

where $\mu_k(t)$ is the content of Memory k, which will be transferred to Memory $(k-1)$ at the end of the current iteration. $\mu_0(t)$ is the initialization of the memory, containing all a priori knowledge about system dynamics. $\alpha(s)$ and $\beta(s)$ are two filters that must be designed so to guarantee convergence.

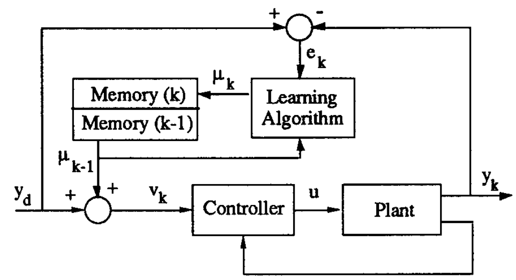

Fig. 1 – Block diagram of learning control

It is easy to verify that μ_k satisfies an iterative equation whose convergence is guaranteed by the contraction condition

$$\big|\beta(j\omega) - \alpha(j\omega)w(j\omega)\big| < 1, \quad (9)$$

where it is assumed that $\alpha(s)$ and $\beta(s)$ are stable, and that the signals are Fourier transformable. This condition generalizes the one in [1], where $\beta \equiv 1$. The limit values, when $k \to \infty$, for the error and the plant input are respectively

$$\begin{aligned} e_\infty(j\omega) &= \frac{\big(1-\beta(j\omega)\big)\big(1-w(j\omega)\big)}{1-\beta(j\omega)+\alpha(j\omega)w(j\omega)}\,y_d(j\omega), \\ v_\infty(j\omega) &= \frac{1-\beta(j\omega)+\alpha(j\omega)}{1-\beta(j\omega)+\alpha(j\omega)w(j\omega)}\,y_d(j\omega). \end{aligned} \quad (10)$$

In the frequency range where $\beta(j\omega) \equiv 1$, these collapse to

$$e_\infty(j\omega) = 0, \qquad v_\infty(j\omega) = \frac{y_d(j\omega)}{w(j\omega)}, \quad (11)$$

showing that an exact learning is obtained for these frequency components, through asymptotic plant inversion.

If β is restricted to be the unity in (8), the convergence condition (9) may not hold. Introduction of a non unitary and frequency-dependent $\beta(j\omega)$ enforces convergence of the learning process, in face of a tracking performance degradatation.

In the implementation of the learning functions α and β, one can usefully resort to some non-causal digital processing, e.g. finite impulse response filters that improve convergence speed and reduce the oscillatory behavior during learning. In particular, in [8–10] Weighted Centered Arithmetic Mean (WCAM) [23] filters of order n were used. If the sampling time is T_c, this filter is characterized by the transfer function

$$W^{(n)}_{WCAM}(s) = \frac{a_0}{2n+1} + \sum_{i=1}^{n} \frac{a_i e^{-siT_c} + a_i e^{siT_c}}{2n+1}, \quad (12)$$

where $W^{(n)}_{WCAM}(j\omega)$ is a real function.

Finally, a further improvement can be obtained through the shift of data in time, advancing them by T_d with the goal of satisfying eq. (9) for a larger set of frequencies. In the Laplace domain, this amounts in introducing an anticipative term e^{sT_d} in the learning filters.

All the above reasoning can be applied to a linearized and decoupled version of a multi-input multi-output nonlinear plant, viz. the robot arm, resulting in a decentralized approach to learning. However, being the synthesis developed on a plant approximation, even when condition (9) is nominally satisfied for each local input-output channel, there still could be stability problems in the real learning process. In practice, the design of learning controllers based on a known linear approximation generally results in satisfactory results [4]. This procedure will be illustrated in the next two sections for a robot with joint elasticity.

4. Design for a Two-Link Arm

We will consider a two-link robot with elasticity at both joints, moving in a vertical plane. The blocks in the general model (1) become (see Fig. 2):

$$\begin{aligned}
M_1(q) &= \begin{bmatrix} a_1 + 2a_2\cos q_2 & a_3 + a_2\cos q_2 \\ a_3 + a_2\cos q_2 & a_3 \end{bmatrix} \\
M_2 &= \begin{bmatrix} 0 & a_6 \\ 0 & 0 \end{bmatrix}, \quad M_3 = \begin{bmatrix} a_4 & 0 \\ 0 & a_5 \end{bmatrix} \\
H_{B1}(q,\dot q)\dot q &= \begin{bmatrix} -a_2 \sin q_2 (\dot q_2^2 + 2\dot q_1 \dot q_2) \\ a_2 \sin q_2 \dot q_1^2 \end{bmatrix} \qquad (13) \\
K_E &= \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \\
e_1(q) &= \begin{bmatrix} a_7 \cos q_1 + a_8 \cos(q_1+q_2) \\ a_8 \cos(q_1+q_2) \end{bmatrix}.
\end{aligned}$$

Here, $H_{A1} = H_{B2} = H_{B3} = 0$. Moreover, no damping is assumed to be present ($F_l = F_m = 0$).

For designing the learning filters and the stabilizing linear feedback, it is convenient (but not mandatory) to perform a linearization of the dynamic equations at an operating point $(q_0, \theta_0, 0, 0)$, with θ_0 related to q_0 through an equation similar to (3). This yields

$$\begin{bmatrix} M_1(q_0) & M_2 \\ M_2^T & M_3 \end{bmatrix} \begin{bmatrix} \Delta\ddot q \\ \Delta\ddot\theta \end{bmatrix} + \begin{bmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta\theta \end{bmatrix} = \begin{bmatrix} 0 \\ \Delta u \end{bmatrix}. \quad (14)$$

K_{ij} will contain both the elasticity matrix K_E and the elements of the matrix $\partial e_1(q_0)/\partial q$. If a linear feedback is closed on the motor variables (see also (2)), then

$$\Delta u = -K_P \Delta\theta - K_D \Delta\dot\theta. \quad (15)$$

The actual values of K_P and K_D should be chosen satisfying some desired steady state response and specified transient behavior.

From (16), one can derive the 2×2 open-loop transfer matrix from $\Delta u(s)$ to $\Delta q(s)$, whose diagonal elements will be used independently for the design of decentralized learning filters $\alpha_i(s)$ and $\beta_i(s)$, $i = 1, 2$.

Indeed, the whole procedure requires the knowledge of the coefficients in the linearized equations (14). However, only approximate values are needed at this stage. On the other hand, one can also perform simple experiments on the real arm in order to tune the gains in (15) and to determine transfer functions from the ith motor torque to the ith link position.

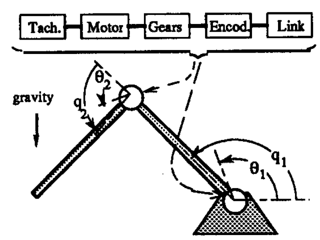

Fig. 2 – A two-link robot with elastic joints

The assumed robot geometric and inertial data (including the motors) provide the following model parameter values:

$$\begin{array}{lll} a_1 = 11.743 & a_4 = 70.656 & a_7 = 189.17 \\ a_2 = 2.7 & a_5 = 23.296 & a_8 = 52.928. \\ a_3 = 2.4817 & a_6 = 0.1456 & \end{array} \quad (16)$$

The elastic coefficients are

$$k_1 = 29800, \qquad k_2 = 14210. \quad (17)$$

With the arm equations linearized at $q_0 = \theta_0 = 0$, be an 'effective' inertia matrix (at the link side of the elastic transmissions) is obtained by adding up rows/columns 1 and 3, and 2 and 4 from the inertia matrix in (14),

$$M_{eff}(0) = \begin{bmatrix} a_1 + 2a_2 + a_4 & a_2 + a_3 + a_6 \\ a_2 + a_3 + a_6 & a_3 + a_5 \end{bmatrix}. \quad (18)$$

The gains in the feedback controller (15) are then designed assuming that 90% of the gravity is mechanically compensated and that a maximum error of 0.001 rad is tolerated in the chosen configuration q_0, i.e. with the arm fully extended. Selecting a damping ratio of $\zeta = 0.6$ leads to

$$K_P = \begin{bmatrix} 24000 & 0 \\ 0 & 5300 \end{bmatrix}, \; K_V = \begin{bmatrix} 1800 & 0 \\ 0 & 450 \end{bmatrix}, \quad (19)$$

resulting in natural frequencies $\omega_{n1} \approx 36$ rad/sec and $\omega_{n2} \approx 30$ rad/sec for the two control loops.

Using only the diagonal elements $w_{ii}(s)$ of the closed-loop transfer matrix $W(s)$, the synthesis of the learning filters is performed on the Nyquist diagrams of $\beta_i(j\omega) - \alpha_i(j\omega)w_{ii}(j\omega)$. This typically requires a trial and error approach, and may take advantage of the designer's classical control experience. In our case, a satisfactory behavior was obtained using:

$$\begin{aligned} \alpha_1(s) &= \frac{10(0.1s+1)(s^2+16s+1600)}{(0.01s+1)(s^2+640s+160000)} e^{0.015s} \\ \beta_1(s) &= \frac{0.00025s+1}{0.001s+1} \left(\frac{1+e^{-sT_c}+e^{sT_c}}{3} \right)^4 \\ \alpha_2(s) &= \frac{2.5(0.1s+1)(s^2+16s+1600)}{(0.01s+1)(s^2+320s+40000)} e^{0.015s} \\ \beta_2(s) &= \beta_1(s). \end{aligned} \quad (20)$$

Here, $T_c = 0.0025$ sec is the actual sampling time used for closed-loop control. Within $\beta_i(s)$, a first order filter $WCAM^{(1)}(s)$ is used repeatedly. Each application produces output samples $y(k)$ that average input data $u(k)$ in time as

$$y(k) = \frac{u(k-1) + u(k) + u(k+1)}{3}. \quad (21)$$

Moreover, a time lead of $T_d = 0.015$ sec is introduced in $\alpha_1(s)$ and $\alpha_2(s)$.

Figures 3 and 4 show the Bode diagrams of the modulus of the local transfer functions $w_{11}(j\omega)$ and $w_{22}(j\omega)$, before (dashed) and after (continuous) the introduction of the learning filters. In these plots, the original transfer functions already include the action of the compensation feedback (19). The associated Nyquist diagrams of $\beta_i(j\omega) - \alpha_i(j\omega)w_{ii}(j\omega)$, $i = 1, 2$, are within the unitary circle for all frequencies; according to (9), convergence of the learning iterations is guaranteed.

5. Simulation Results

The two-link robot with elastic joints specified by the set of parameters (16) and (17) was used for simulation. Learning control was tested on a quintic polynomial trajectory

$$\frac{q_d(t) - q_0}{q_f - q_0} = 6(\frac{t}{t_f})^5 - 15(\frac{t}{t_f})^4 + 10(\frac{t}{t_f})^3, \quad (22)$$

satisfying zero initial and final velocity and acceleration, and chosen equal for both links. The traveling distance and time are set to $q_f = 1$ rad and $t_f = 1$ sec, resulting in a quite fast motion. The robot starts at rest from $q_0 = 0$, with no initial joint deformation and in the maximum gravity loading configuration. The final position implies a non-zero joint deformation.

The system equations were integrated using a 2nd order Runge-Kutta method with sampling time $T_s = 0.0025$ sec, and the tracking results using learning are reported in Figs. 5–8 for a total of 1.5 sec.

It can be seen that the tracking accuracy is very good after about 50 iterations. However, the error is already reduced by a factor of 25 in about 18 iterations, from a peak value of 0.5 rad at the first trial when only motor PD control and reference trajectory feedforward are applied; in particular, it is halved in just 9 trials (Figs. 6 and 8). For a rough comparison, the learning method in [19] achieves similar results in double the number of iterations (over 100).

The memory μ_k, shown in Fig. 9 for joint 1, learns the necessary modification to the trajectory position reference (see eq. (8)), so to improve the output tracking performance. Accordingly, the input motor torques are corrected over trials, progressively changing their composition from feedback to feedforward action, as illustrated in Fig. 10 for joint 1. Note that in the early iterations of the learning process, oscillations grow up around the final configuration (error, memory, and input torque plotting scales are magnified there by a factor 50). The large transient errors during the gross motion dominate the whole learning process at this stage. As soon as these are reduced, also the terminal part of the error will be smoothed. Small torque oscillations after 1 sec are needed to keep the links at their desired position, counteracting with the motor action the residual undamped vibration. These oscillations are indeed eliminated in the output tracking errors (i.e. on the link side) of Figs. 6 and 8.

6. Conclusions

The design of a repetitive learning controller for robots with elastic joints has been presented, following a frequency-domain approach. This choice allows to properly handle the presence of high-frequency dynamics associated to joint elasticity, which would exclude the application of other iterative learning algorithms.

The frequency shaping of the method does only affect the bandwidth where the desired behavior is exactly reproduced. Convergence properties are preserved without the need of using a slower two-stage betterment process. The numerical results obtained in simulation for a fully nonlinear multi-link case show that our algorithm is superior to the only other existing learning approach to this problem.

The present learning scheme is easy to be implemented and requires on-line feedback of motor measures only. The simplicity gained with learning control is particularly relevant for robots with elastic joints, when compared with the real-time complexity of model-based techniques producing the same tracking accuracy. As a result, the choice of iteratively inverting the system should be considered as computationally advantageous in this case, even when the robot dynamics is relatively well known.

Acknowledgements

This paper is based on work supported by the *Consiglio Nazionale delle Ricerche*, contract no. 91.01946.PF67 (*Progetto Finalizzato Robotica*).

References

[1] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *J. of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.

[2] T. Omata, S. Hara, and M. Nakano, "Nonlinear repetitive control with application to trajectory control of manipulators," *J. of Robotic Systems*, vol. 4, no. 5, pp. 631–652, 1987.

[3] S. Arimoto, T. Naniwa, and H. Suzuki, "Selective learning with a forgetting factor for robotic motion control," *1991 IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, pp. 728–733, Apr. 1991.

[4] S. Arimoto, S. Kawamura, and F. Miyazaki, "Convergence, stability, and robustness of learning control schemes for robot manipulators," *Int. Symp. on Robot Manipulators: Modeling, Control, and Education*, Albuquerque, NM, pp. 307–316, Nov. 1986.

[5] C. Cosner, G. Anwar, and M. Tomizuka, "Plug in repetitive control for industrial robotic manipulators," *1990 IEEE Int. Conf. on Robotics and Automation*, Cincinnati, OH, pp. 1970–1975, May 1990.

[6] C.G. Atkeson and J. McIntyre, "Robot trajectory learning through practice," *1986 IEEE Int. Conf. on Robotics and Automation*, S. Francisco, CA, pp. 1737–1742, Apr. 1986.

[7] A. De Luca, G. Paesano, and G. Ulivi, "A frequency-domain approach to learning control of robots," *4th IEEE Int. Symp. on Intelligent Control*, Albany, NY, pp. 66–70, Sep. 1989; to appear on *IEEE Trans. on Industrial Electronics*.

[8] A. De Luca and F. Mataloni, "Learning control for redundant manipulators," *1991 IEEE Int. Conf. on Robotics and Automation*, Sacramento, CA, pp. 1442–1450, Apr. 1991.

[9] M. Poloni and G. Ulivi, "Iterative trajectory tracking for flexible arms with approximate models," *5th Int. Conf. on Advanced Robotics*, Pisa, I, pp. 108–113, Jun. 1991.

[10] M. Poloni and G. Ulivi, "Iterative learning control of a one-link flexible manipulator," *Prepr. 3rd IFAC Symp. on Robot Control*, Vienna, A, pp. 273–278, Sep. 1991.

[11] M.C. Good, L.M. Sweet, and K.L. Strobel, "Dynamic models for control system design of integrated robot and drive systems," *ASME J. of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 53–59, 1985.

[12] M. Spong, "Modeling and control of elastic joint robots," *ASME J. of Dynamic Systems, Measurement, and Control*, vol. 109, pp. 310–319, 1987.

[13] S.H. Lin, S. Tosunoglu, and D. Tesar, "Control of a six-degree-of freedom flexible industrial manipulator," *IEEE Control Systems Mag.*, vol. 11, no. 3, pp. 24–30, 1991.

[14] T.J. Tarn, A.K. Bejczy, A. Isidori, and Y. Chen, "Nonlinear feedback in robot arm control," *23rd IEEE Conf. on Decision and Control*, Las Vegas, NV, pp. 736–751, Dec. 1984.

[15] A. De Luca, "Dynamic control of robots with joint elasticity," *1988 IEEE Int. Conf. on Robotics and Automation*, Philadelphia, PA, pp. 152–158, Apr. 1988.

[16] L. Lanari and J.T. Wen, "A family of asymptotically stable control laws for flexible robots based on a passivity approach," *5th Int. Conf. on Advanced Robotics*, Pisa, I, pp. 102–107, Jun. 1991.

[17] F. Ghorbel, J.Y. Hung, and M. Spong, "Adaptive control of flexible-joint manipulators," *IEEE Control Systems Mag.*, vol. 7, no. 6, pp. 9–13, 1987.

[18] B. Brogliato and R. Lozano Léal, "Adaptive control of robot manipulators with flexible joints," *1991 American Control Conf.*, Boston, MA, pp. 938–943, Jun. 1991.

[19] F. Miyazaki, S. Kawamura, M. Matsumori, and S. Arimoto, "Learning control scheme for a class of robots with elasticity," *25th IEEE Conf. on Decision and Control*, Athens, GR, pp. 74–79, Dec. 1986.

[20] P. Tomei, "A simple PD controller for robots with elastic joints," *IEEE Trans. on Automatic Control*, vol. 36, pp. 1208–1213, 1991.

[21] S.H. Murphy and J.T. Wen, "Simulation and analysis of flexibly jointed manipulators," *29th IEEE Conf. on Decision and Control*, Honolulu, HI, pp. 545–550, Dec. 1990.

[22] A. De Luca, "Nonlinear regulation of robot motion," *1st European Control Conference*, Grenoble, F, pp. 1045–1050, Jul. 1991.

[23] N.B. Jones (Ed.), *Digital Signal Processing*, IEE Control Engineering Series no. 22, Peter Peregrinus, UK, 1982.

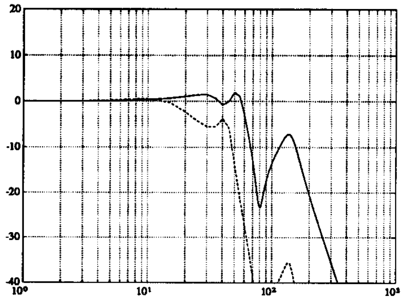

Fig. 3 – $|w_{11}(j\omega)|$ before (dashed) and after filtering

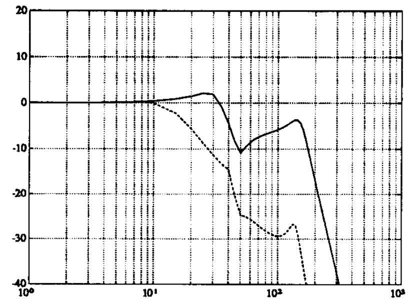

Fig. 4 – $|w_{22}(j\omega)|$ before (dashed) and after filtering

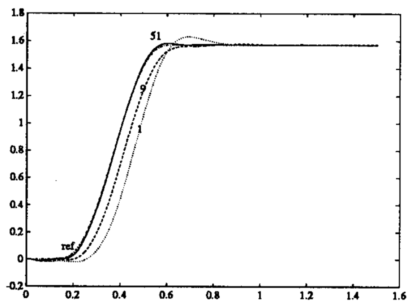

Fig. 5 – Trajectory tracking for link 1 at $k = 1, 9, 51$

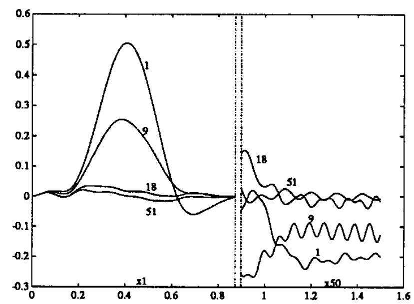

Fig. 6 – Tracking error for link 1 at $k = 1, 9, 18, 51$

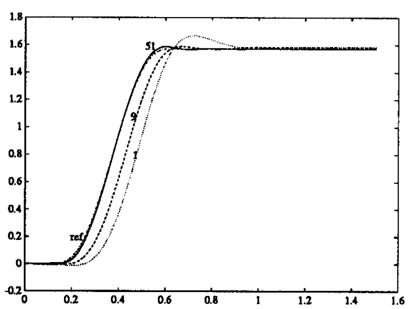

Fig. 7 – Trajectory tracking for link 2 at $k = 1, 9, 51$

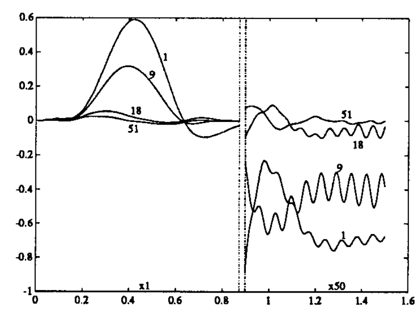

Fig. 8 – Tracking error for link 2 at $k = 1, 9, 18, 51$

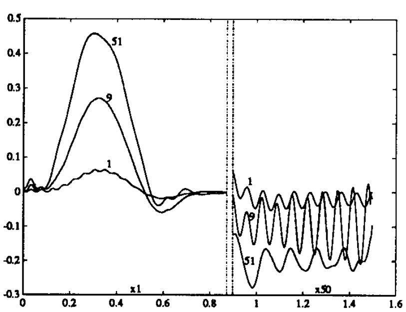

Fig. 9 – Memory for joint 1 at $k = 1, 9, 51$

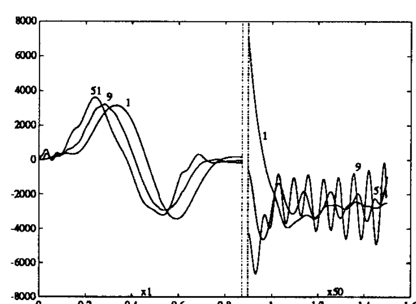

Fig. 10 – Input torque for joint 1 at $k = 1, 9, 51$