N89 - 10105

PERFORMANCE IMPROVEMENT OF ROBOTS USING A LEARNING CONTROL SCHEME

Ramuhalli Krishna
Advanced Technology & Research, Inc.,
3933 Sandy Spring Road
Burtonsville, MD 20866

and

Pen-Tai Chiang and Jackson C.S. Yang
Robotics Laboratory
Department of Mechanical Engineering
University of Maryland
College Park, MD 20742

ABSTRACT

Many applications of robots require that the same task be repeated a number of times. In such applications, the errors associated with one cycle are also repeated every cycle of the operation. An off-line learning control scheme is used here to modify the command function which would result in smaller errors in the next operation. The learning scheme is based on a knowledge of the errors and error rates associated with each cycle. Necessary conditions for the iterative scheme to converge to zero errors are derived analytically considering a second order servosystem model. Computer simulations show that the errors are reduced at a faster rate if the error rate is included in the iteration scheme. The results also indicate that the scheme may increase the magnitude of errors if the rate information is not included in the iteration scheme. Modification of the command input using a phase and gain adjustment is also proposed to reduce the errors with one attempt. The scheme is then applied to a computer model of a robot system similar to PUMA 560. Improved performance of the robot is shown by considering various cases of trajectory tracing. The scheme is also applied to a real robot PUMA 560. The results show that the proposed scheme can be successfully used to improve the performance of actual robots within the limitations of the repeatability and noise characteristics of the robot.

1.0 INTRODUCTION

Several methods of performance improvement of robots have been attempted previously by many researchers. Some of these employ on-line adaptive control schemes[1,2] considering factors such as flexibility of the arms and variation of loads. Such control schemes are required for trajectories which are not defined a priori or require accurate trajectory control in the first attempt. For repetitive type of operations, as most commonly required in industrial applications, such as welding, cutting or sealing, a control scheme which can learn based on its previous performance appears to be more attractive because of the simplicity of the technique. A method to obtain a modified command input signal called Computed Repetitive Adjustment Technique (CREATE)[3] is embodied in an algorithm for correcting a robot's motion in successive testing of the same job. This repetitive testing is continued until the trajectory errors are within the acceptable bound before performing the actual work. The technique was later applied[4,5] to improve the performance of a mathematical model of a three-link robot arm.

A scheme very similar to CREATE was studied by Craig[6] for application by considering a linearized model of the robot. Conditions for the convergence of the scheme to yield minimum errors were obtained with the assumption that several critical parameters of the system are known. Arimoto[7] proposed a learning scheme based on measuring error rate only. He later extended[8,9] the control scheme to include the error and error rate information. However, the conditions[8] he arrived at are found to be unsatisfactory. Several other researchers[10,11,12] have attempted to obtain convergence conditions for the iteration scheme, but the analysis are generally inadequate. Togai[13,14] obtained interesting algorithms by using discrete analysis and by applying optimal control techniques. An algorithm based on optimal control technique is also given by Harokopos[15] for continuous systems. Bedewi[16] used CREATE technique to refine the performance of the robot starting with a dynamic inverse of the model of the system. In this paper, the learning control scheme proposed by Arimoto[8] will be considered. Analysis of the scheme will be attempted with a servosystem model. The scheme will then be applied to a mathematical model of a three-link robot to show the improvement in performance.

2.0 SECOND ORDER SERVOSYSTEM MODEL

2.1 Description of the Servosystem Model

A servo control design is given in this section which will be applied to design independent control of each joint of the robot. The second order servosystem model is also used in this section to test the proposed learning control scheme. The scheme is then applied to improve the performance of the robot model in section 3.

The dynamics of the servosystem can be represented, by ignoring the damping factor for the sake of simplicity, as

$$I\ddot{\theta} = T_c \tag{1}$$

where θ is the angular position (a function of time, t)

T_c is the control torque

I is the moment of inertia of the servo, and

$\ddot{\theta}$ is the second derivative of θ with respect to time

It is necessary to feedback the angular position, θ, and its rate, $\dot{\theta}$, to obtain a stable position control system. An integral feedback of the position may also be used to obtain a higher speed of response. However, an integral feedback increases the order of the system and introduces oscillations in the system response. These characteristics are not desirable for the present application and hence only a proportional plus derivative controller is considered for stabilizing the servosystem as shown in Fig. 1. The characteristic equation of the closedloop system can be written as,

$$s^2 + (K_2/I)s + (K_1/I) = 0 \tag{2}$$

This equation may be compared with the damped oscillatory system

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = 0 \tag{3}$$

where ζ is the damping ratio and ω_n is the natural frequency of the system. It is desirable to have a high value of ω_n to get a large bandwidth of the system. The stability of the second order system is guaranteed as long as K_1 and K_2 remain positive. However, the feedback gain value K_1 can not be increased beyond a certain a limit (due to the practical limitations of the actuator) in an effort to increase the bandwidth of the system. Within these limits, the feedback gain values are obtained using the relations (obtained from Eqs. (2) and (3)) as,

$$K_1 = \omega_n^2 I \quad \text{and} \quad K_2 = 2\zeta\omega_n I \tag{4}$$

The rate feedback gain, K_2, is assumed to achieve critical damping in the closedloop system. For such a case, the system will have the highest speed of response corresponding to the lowest settling time and for a given bandwidth of the system. Fig. 2 shows the transient response of the second order system for a step input and for the selected feedback gain values.

Case			
(a)	$K_1 = 100$	$K_2 = 20$	$\omega_n = 10$ rad/sec
(b)	$K_1 = 144$	$K_2 = 24$	$\omega_n = 12$ rad/sec
(c)	$K_1 = 625$	$K_2 = 50$	$\omega_n = 25$ rad/sec

Case (c) shows the lowest rise time of approximately 0.2 sec. The same feedback gain values will be considered later to control the robot arm joints.

2.2 A Learning Control Scheme

The second order system considered may be represented as (from Fig. 2),

$$(Is^2 + K_2 s + K_1)\theta = \theta_c \tag{5}$$

3

l, the command function, θ_C, is the same as the desired trajectory, θ_C, is

$$\dot\theta_d - \dot\theta_a) + b(\theta_d - \theta_a) \tag{6}$$

re constants proportional to the error rate and the error, respectively. For
i the command signal, θ_C , is modified as,

$$\theta_c + \theta_e \tag{7}$$

the system to this modified command signal is determined. The new error
obtained using Eq. (6) The command signal is again modified according to
w command signal is used in Eq. (2.5) and the response of the system is
: process is repeated until the errors, θ_e, fall within an acceptable value.

idy is possible for the second order system considered here. Eq. (5)
second order system is written in a slightly different form for the kth

$$\theta_k = \theta_{ck} \tag{8}$$

eration the equation as

$$ + K_1\theta_{k+1} = \theta_{ck} + \theta_{ek} \tag{9}$$

$$_k) + b(\theta_d - \theta_k) \tag{10}$$

(8) from Eq. (9), we get

$$= \theta_{ek} \tag{11}$$

$$_k$$

$$a\dot d_k + bd_k \tag{12}$$

$$(\tau)\, d\tau \tag{13}$$

4

Then

$$J_{k+1} = \int_0^t \theta^2_{ek+1}(\tau)\, d\tau$$

$$= \int_0^t \left[\theta_{ek} - (a\dot{d}_k + bd_k)\right]^2 d\tau$$

Expanding and using equations (13) and (10), we get

$$J_k - J_{k+1} = \int_0^t \{(2aq - a^2 - 2b)\,\dot{d}_k{}^2 + (2bp - b^2)\,d_k{}^2\}\, d\tau + a\dot{d}_k{}^2(t)$$

$$+ (ap + bq - ab)\, d_k{}^2 + 2bd_k\dot{d}_k$$

$$\geq 0, \text{ for } a^2 > 2b$$
$$a < 3q/2$$
$$b < 2p \text{ and for small error rates} \qquad (14)$$

Eq.(14) gives sufficient conditions for the guaranteed convergence of the error to zero value with each step of the proposed iteration scheme.

Since only the command signals are being modified, the torque requirements on the actuators will not drastically change provided the initial errors are small. The feedback values selected for the application in this section are $K_1 = 100$ and $K_2 = 20$ and corresponds to a closedloop frequency of 10 rad/sec. The response of the servosystem to a command input,

$$\theta_C = \sin(\omega t)$$

is shown in Fig. 3 for $\omega = \pi$ rad/sec. The actual trajectory shows a phase lag and an amplitude modification. Application of the learning control scheme shows reduced errors with the first iteration (Fig. 4) and the third iteration (Fig. 5) and for the parameters a= 0.1 and b= 0.9. The figures compare average errors associated with each iteration. The average

as

$$AE = \frac{\sum_{i=1}^{NP} | \theta_{ei} |}{NP} \tag{15}$$

mber of points on the trajectory. Higher values of ω were considered next
ance of the iteration scheme was studied. Fig. 6 shows the average error
each iteration for various values of "a" and "b" and for ω = 2 π rad/sec.
rly show that for the case in which a = 0, the errors tend to increase after a
n accordance with the derived conditions (Eq. (14)). Similar results are
for ω =4 π rad/sec. In general, the results show that the rate of
creases with increasing values of "a" up to a certain value. For the present
best results are obtained with a= 0.2 and b= 0.8 at all values of ω. The
ing error rate in the modification procedure is obvious from Figs. 6 and 7.
bserved that the conditions given by Eq. (14) are rather conservative

1 Phase Adjustment Technique

he previous results that there is a unique command function, θ_c , for
trajectory, θ_a , will be the same as the desired trajectory, at least for
single-output systems. In the previous examples, inclusion of the error
helped a great deal in arriving at the unique command function with only a
The next obvious question is whether it is possible to obtain the ideal
on with a single iteration so that the resulting trajectory would follow the
y. An attempt has been made in this section to answer the question.

of the second order system shows that the output of the system follows the
ly, provided the frequencies associated with the function are well within the
th. As the input frequency approaches the system bandwidth, the output of
s marked deviation from the input signal. The input-output relationship is
se and the gain plots in which the phase difference and the ratio of the
output to the input are plotted as a function of the input frequency.

the previous example, θ_d = sin(ωt), with ω = π rad/sec. The time
system is as shown in Fig. 3., for the first trial. A modification scheme
was employed with a=0.1 and b=0.9. The actual trajectories obtained
on and third iteration are shown in Figs. 4 and 5, respectively. From Fig. 3
ues of phase (-0.5969 rad) and gain (0.91) were obtained. The command
dified as

1 sin(ωt + 0.5969)

tory obtained for this modified command signal is shown in Fig. 8. The
and the desired trajectory overlap within the accuracy of the plotter. Thus,
linear system, it was possible to obtain a modified command signal to
ed trajectory without an iteration scheme. This method will be applied to
del of three-link robot system in section 3, which is an example of a
ic system.

6

3.0 APPLICATION TO A THREE-LINK ROBOT

3.1 Dynamics of a Three-Link Robot Arm

Fig. 9 shows the selected coordinate reference frame for the three-link robot arm. X, Y, Z, is the inertial system with the origin at joint 1. The fourth coordinate frame has the origin fixed at the tip of the robot arm. The first link can rotate only about the vertical axis Z, carrying the second and the third links. The second link can rotate about an axis fixed in link 2 and is normal to both the first and the second link. The third link moves about an axis parallel to the axis of rotation of the second link.

The three transformations are represented by the following matrices

$$
\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix}
$$

$$\text{or} \quad r = [A_1] r_1 \tag{16}$$

$$
\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_2 & -S_2 & C_2 d_2 \\ 0 & S_2 & C_2 & S_2 d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \\ 1 \end{bmatrix}
$$

$$\text{or} \quad r_1 = [A_2] r_2 \tag{17}$$

$$
\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C_3 & -S_3 & C_3 d_3 \\ 0 & S_3 & C_3 & S_3 d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_3 \\ Y_3 \\ Z_3 \\ 1 \end{bmatrix}
$$

$$\text{or} \quad r_2 = [A_3] r_3 \tag{18}$$

where $C_j = \cos \theta_j$, and $S_j = \sin \theta_j$, j=1,2,3
A point r_i described with respect to link i can be related to the base coordinate by

$$r = [T_i] r_i \tag{19}$$

where, $T_1 = [A_1]$, $T_2 = [A_1][A_2]$, and $T_3 = [T_2][A_3]$

The dynamic equation for the robot arm is obtained as, [17]

$$F_i = \sum_{j=1}^{3} D_{ij}\ddot{q}_j + I_{a_i}\ddot{q}_i + \sum_{j=1}^{3}\sum_{k=1}^{3} D_{ijk}\dot{q}_j\dot{q}_k + D_i \qquad (20)$$

where:

$$D_{ij} = \sum_{p=\max i,j}^{3} \text{Trace}\left(\frac{\partial T_p}{\partial q_j} J_p \frac{\partial T_p^T}{\partial q_i}\right)$$

$$D_{ijk} = \sum_{p=\max i,j}^{3} \text{Trace}\left(\frac{\partial^2 T_p}{\partial q_j \partial q_k} J_p \frac{\partial T_p^T}{\partial q_i}\right)$$

$$D_i = \sum_{i=1}^{3} -m_p\, g^T \frac{\partial T_p^T}{\partial q_i} r_p$$

I_{a_i} is the Ith actuator inertia, m_p is the mass of the pth link
g is a vector in the direction of the gravity force
J_p is a pseudo inertia matrix for the pth link

Eq. (20) are used to compute the joint torques required. For the purpose of simulation of the first, second and the third links are assumed to have lengths equal to 0.5m, 0.4m, and masses 4kg, and 2 kg, and 1kg, respectively. The inertias of the links 1,2, and, 3 are assumed to be 5×10^{-4}, and 2.5×10^{-4}, and 1.25×10^{-4} kgm^2, respectively. Actuator inertias corresponding to joints 1,2, and, 3 are taken to be 1×10^{-3}, 5×10^{-3}, and 2.5×10^{-4}, kgm^2 respectively.

3.2 Inverse Kinematics

The trajectory to be traced by the robot arm is given by a set of points which are usually defined in the base (inertial) coordinate system. Therefore, it is necessary to find the joint angles corresponding to each point on the trajectory using the geometry of the robot. Let us assume that the tip of the robot arm follows the given trajectory, so that a transformation of the tip point into the base coordinates can be made as follows:

$$(P) = [T_3] (r_3) \qquad (21)$$

where r_3 is actually the origin of the fourth coordinate system. Substituting for T_3 and expanding we get,

$$P_x = -S_1(C_2D_2 + C_{23}D_3) \qquad (22)$$
$$P_y = C_1(C_2D_2 + C_{23}D_3) \qquad (23)$$
$$P_z = D_1 + D_2S_2 + D_3C_{23} \qquad (24)$$

Algebraic manipulations of the above equations give the expressions for the joint angles as

$$\theta_1 = -\tan^{-1}(x_1/y_1) \qquad (25)$$

8

$$\frac{-d_1\}}{d} - \cos^{-1}\{ \frac{d^2 + d_2{}^2 - d_3{}^2}{2dd_2} \} \qquad (26)$$

$$\frac{{}_1/c_1)^2 + (z_1 - d_1)^2 - d_2{}^2 - d_3{}^2}{2d_2d_3} \} \qquad (27)$$

ompute the joint angles only once in order to find the command signals
ry. This can be done off-line using the robot central computer in the
iterative technique is then applied to modify the joint angle commands to
gle errors to zero.

rol system

e employed here is to control the motion of the joints so that the tip of the
trajectory. Independent control of each of the joints based on only the
position and rate feedbacks is considered here for the sake of simplicity.
ied in the base coordinate system is converted into joint command angles
kinematics. The dynamic equations (20) are rearranged for the
ol techniques as follows.

$$\sum_{k=1}^{3} D_{ijk}\dot{\theta}_j\dot{\theta}_k - \sum_{\substack{j=1 \\ j\neq 1}}^{3} D_{ij}\ddot{\theta}_j - D_i\} / (D_{ii} + I\theta_i) \qquad (28)$$

int angular coordinates, T_{ci} is the torque about the joint i. Control
ed from the block diagram shown in Fig.1 as

$$\langle_1(\theta_i - \theta_c) \qquad (29)$$

a sixth order nonlinear differential equation with coupled coefficients.
numerically solved here using a fourth order Runge-Kutta integration

of Learning Control Scheme

the case in which the desired trajectory for the tip of the arm is a circle.
ined in the inertial coordinate system as follows.

$$s(\omega t)$$

$$(30)$$

$$0.5 \sin (\omega t)$$

am is modified so that the joint command angles are obtained from the
ystem through a kinematic transformation. The computer time required
approximately 20 minutes and hence only four iterations are considered
The results obtained are shown in Fig. 11. There are two curves in
ling to the errors associated with X and Z coordinates. The errors in

9

the Y direction is not so important and is not shown here. It may be seen that the errors tend to become smaller at a faster rate when the error rate information is included in the command input modification procedure. Fig. 12 shows the torque characteristics of the joints 1 and 2 before and after the application of the learning control scheme. Figs. 13 to 15 show the actual response of the robot corresponding to a=0 and b=1. A trajectory very close to the desired circular trajectory could be obtained with a=0.1 and b=1 after 6 iterations (figure not shown).

3.5 Application of Gain and Phase Adjustment Technique

It was shown in section 2.3 that a modified command input could be obtained so that the resulting actual trajectory is very close to the desired trajectory for linear dynamic systems. However, the robots are in general are characterized by nonlinearities and varying moments of inertia parameters. Coupling between the joint coordinates are also very significant. Two examples of trajectory tracing, one a circle and the other a straight line, is considered for the application of the gain and phase modification technique.

The trajectory defined by Eq.(30) describes a circle in the X-Z plane. The actual trajectory traced by the robot is given in Fig. 13. The responses of the robot after first and fourth iterations using the learning scheme are as shown in Figs. 14 and 15, respectively. It may be seen that even after four iterations the trajectory traced is still not a complete circle. The gain and phase modification technique was then attempted as follows.

The response of the system to the command trajectory which is the same as the desired trajectory was plotted in the X and Z coordinate system as a function of time (not shown). Ideally, X coordinate should be a cosine function and Z a sine function as given by Eq. (30). However, depending on the value of w, there will be a phase and gain change. The actual response was found to fit the following functions.

$$X = \cos(\omega t - 0.17)$$
$$Z = \sin(\omega t - 0.19)$$
(31)

Modified command signals were then obtained as

$$X = \cos(\omega t + 0.17)$$
$$Z = \sin(\omega t + 0.19)$$
(32)

The response of the robot to the modified command signal is as shown in Fig. 16. The actual trajectory obtained is seen to be very close to the desired trajectory. It is very interesting to see that the nonlinearity, the coupling between the joint coordinates, gravity and the varying moments of inertias did not have much effect as far as the phase shift and gain values are considered. The present example, however, is very simple because the desired trajectory could be represented by simple sine and cosine functions.

The second example considered is the one in which the tip of the robot traces a straight line. The joint angles will have to go through a nonlinear motion so that the tip of the robot could result in a straight line. Hence, this example is more complicated than the first case. The command function in the inertial coordinate system for the example of a straight line is taken to be

$$X = 0.5 - 0.5 \, w \, t$$

10

Y

Z

The respo: system with $w = 1$ is as shown in Fig. 17. For the sake of simplicity, only the Xite is considered for modification. Learning control scheme is then used to obtain arajectory after 6 iterations as shown. For the application of gain and phase mods, the command signal for the X coordinate is obtained as follows.

$$X = 0.5 \, wt + X_e(t + t_p)X_g$$

where X_e ior obtained as

$$X_e \quad X_a$$

and t_p is theshift obtained from the first trail. The response obtained with this modified col is shown in Fig. 17. The phase t_p is taken to be 0.18 secs and the gain, X_g, 1. trial and error). The actual trajectory obtained for this command is almost as gohe trajectory obtained through 6 iterations.

The phase, t_j gain, X_g, were also obtained by a Fourier analysis of the error function associated w: trajectory. Fig. 18 shows the response of the corresponding modified command ($w = 0.2$ secs and $X_g = 1.117$). The response appears to be at least as good as the nse shown in Fig. 17. Hence, it may be concluded that the Fourier analysis can bd to compute the phase and gain value to obtain a modified command and to reducerrors considerably in the second trial. Further minimization of errors may be done ¦ the iterative technique, if necessary.

3.6 Applical to a Real Robot

The performarmprovement technique is attempted on a real robot PUMA 560. Fig. 19 shows a schem: diagram of the robot. Instructions to the robot arm is given by using the operating systeoftware, VAL-II[18]. Different values of the coefficient of the error are considered (withe parameter 'a'=0) to study the rate of convergence of the scheme to yield minimumrors. Fig. 20 shows the performance of the robot after 11 iterations for a desired trajectorof the robot to move along the Y axis at approximately 1.5ft/sec. The figure shows therror in the X direction with the first attempt and after 11 iterations with b=0.2. The erro: associated with the first trial is 0.4145mm and , after 11 iteractions , 0.07mm.

The same case was then repeated with various values of the error coefficient, b, and the average errors obtained as a function of the trial number are shown in Fig. 21. The results clearly show that the rate at which the average error decreases is proportional to the value of b. However, larger values of b tends to increase the average error after a few iterations. This trend was also observed and discussed in section 2.

The iterative technique was also applied to improve the performance of the robot under a variety of other operating conditions. The results show, in general, that the lower limit of the average error after reached (which is 0.1mm) is restricted by the repeatability and the noise associated with the joint sensors. Further work is continuing at this time to include the rate information in the application of the learning scheme to the real robot.

4.0 CONCLUSIONS

An off-line learning control scheme is analyzed here which can be used to improve the performance of robots. Necessary conditions for the iterative scheme to reduce errors with each iteration are derived considering a second order servo system model. The scheme is also applied to a mathematical model of a three-link robot similar to PUMA 560. In general, the results show that the learning control scheme reduces errors more efficiently if the error rate information is included in the scheme. The results also show that the scheme may increase the magnitude of the errors, if the rate information is not included in the iteration scheme. Preliminary results of the application of the technique to a real robot has shown that the scheme can be successfully used to improve the performance of actual robots within the limitations of the repeatability and noise characteristics of the robots.

ACKNOWLEDGMENTS

REFERENCES

1. Dubowsky, S., and Des Forges,D.T., 'The Application of Model-Referenced Adaptive Control to Robotic Manipulators,' Transactions of the ASME, Journal of Dynamic Systems, Measurement and Control, Sep.. 1979, Vol. 101, pp 193-200.

2. Nelson, W.L., Mitra, D., and Boie, R.A., "End Point Sensing and Load Adaptive Control of a Flexible Robot arm," Proceedings of 24th Conference on Decision and Control, Ft. Lauderdale, Florida, December, 1985.

3. "Optimization of Robot Manufacturing Operations," Report to NSF by Advanced Technology and Research, Inc., April, 1983.

4. Bedewi, N.E., "Computed Reference Repetitive Adjustment Technique for a Robot Arm Control in Three Dimensional Space," M.S. Thesis, Department of Mechanical Engineering, University of Maryland, 1984.

5. "Development of Robot Performance Improvement Technique Through Repetitive Testing" Phase II 6 Months Report to NSF by Advanced Technology and Research, Inc., April 1985, Contract No. ECS 8313834.

6. Craig, J.J., "Adaptive Control of Manipulators Through Repeated Trials," American Control Conference, San Diego, June 1984.

7. Kawamura, S., and Miyazaki, F., and Arimoto, S., "Applications of Learning method for Dynamic Control of Robot Manipulators," Proceedings of 24th Conference on Decision and Control, Ft. Lauderdale, Florida, December, 1985.

8. Arimoto, S., Kawamura, S., and Miyazaki, F., "Bettering Operation of Dynamic Systems by Learning: A New Control Theory for Servomechanism or Mechatronic Systems, "Proceedings of 23rd Conference on Decision and Control, Las Vegas,

:cember, 1984, pp. 1064-69.

, "Learning Control Theory for Dynamical Systems," Proceedings of 23rd
on Decision and Control, Ft. Lauderdale, Florida, December, 1985, pp

mata, T., and Nakano, M., "Synthesis of Repetitive Control Systems and
ion," Proceedings of 24th Conference on Decision and Control, Ft.
, Florida, December 1985, pp 1387-92.

d Kato, E., "Iterative Control and Its Application to Motion Control of
- A Direct Approach to Servo Problems," Proceedings of 24th Conference
ı and Control, Ft. Lauderdale, Florida, december 1985, pp 1393-98.

ınd Yamakita, M., "Iterative Generation of Optimal Input of A
r," IEEE Conference on Robotics and Automation, April 1986, pp 579-584.

ınd Yamano,O., "Analysis and design of an Optimal Learning Control
Industrial Robots," Proceedings of 24th Conference on Decision and
Lauderdale, Florida, December 1985, pp 1399-1404.

ınd Yamano, O., "Learning Control and Its Optimality: Analysis and Its
to Controlling Industrial Robots," Proceedings of the IEEE Conference on
d Automation, April 1986, pp 248-253.

E.G., "Optimal Learning Control of Mechanical Manipulators in Repetitive
roceedings of the IEEE Conference on Robotics and Automation, April
)6-401.

E., Chiang, N.E., and Yang, J.C.S., "Robot Position Accuracy
ıt Through Repeated Trials," Paper No. 86-DET-44, Design Engineering
:onference, Columbus, Ohio, October 5-8, 1986.

'Robot Manipulators: Mathematics, Programming and Control," The MIT
achusetts, 1981.

User's Guide to VAL" Unimation, Inc., Danbury, Connecticut.

(a)

(b)
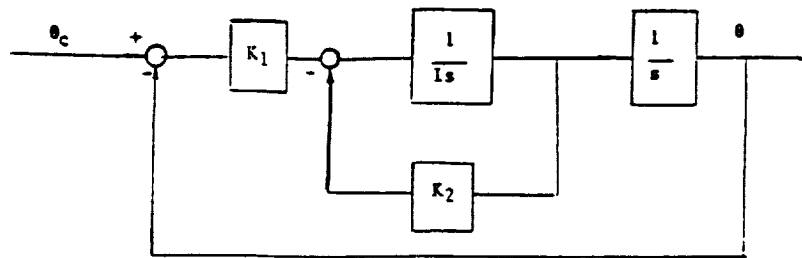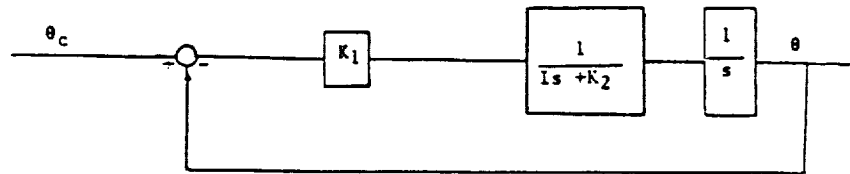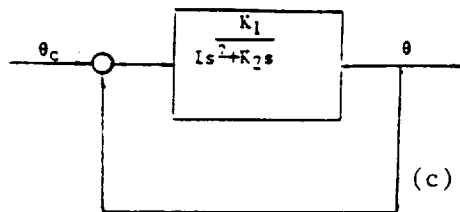
Closed-Loop Transfer Function:

$$\frac{\theta}{\theta_c} = \frac{K_1}{Is^2 + K_2 s + K_1}$$

(c)

Block-Diagram Representation of the Servo Control System and (c) Block-Diagram Reduction Steps
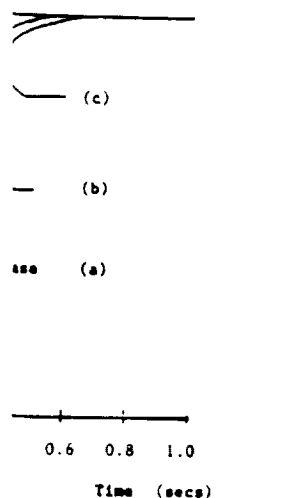
(c)

(b)

(a)

0.6   0.8   1.0

Time (secs)

nput Response for
ent sets of Gain

Actual Trajectory

Desired Trajectory

Average Error = 0.35

Time (secs.)

Fig. 3.   Transient Response of the Servosystem

14

1.0

Actual Trajectory

Desired Trajectory

θ (rad)

Average Error = 0.11

-1.0

2.0

Time (secs.)
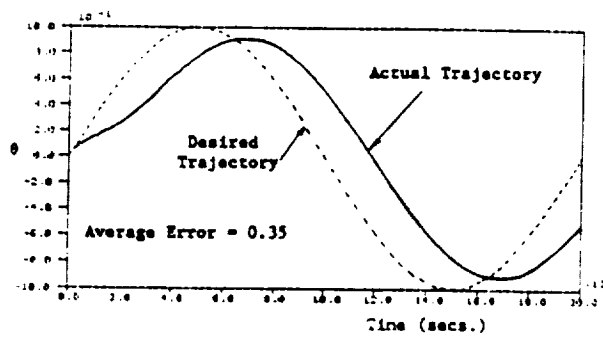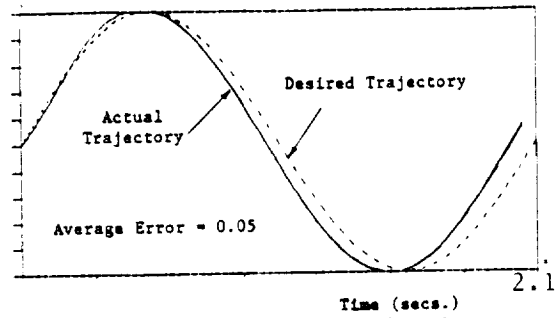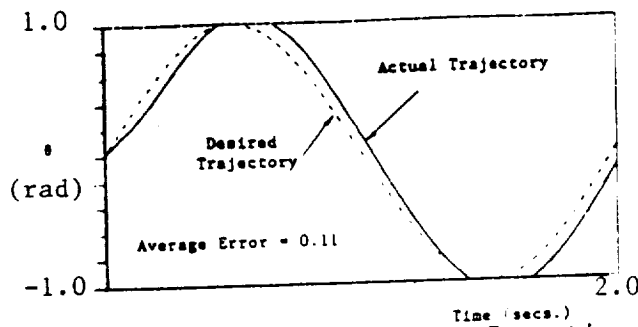
Fig. 4 Response After First Iteration

Desired Trajectory

Actual Trajectory

Average Error = 0.05

2.1

Time (secs.)

Fig. 5. Response After Third Iteration

0.4

a=0.0 b=1.0

AE (rad)

(a)

0 2 4 6 8 10

Number of Iterations

0.4

a=0.0 b=0.9

AE (rad)

(b)

0 10

Number of Iterations

0.4

a=0.1 b=1.0

AE (rad)

(c)

0 10

0.4

a=0.1 b=0.9

AE (rad)

(d)

0 10

0.4

a=0.2 b=0.8

AE (rad)

(e)

0 10

0.4

a=0.3 b=0.7

AE (rad)

(f)

0 10
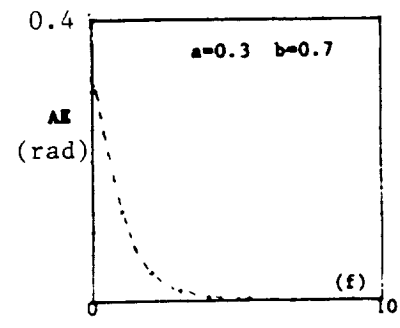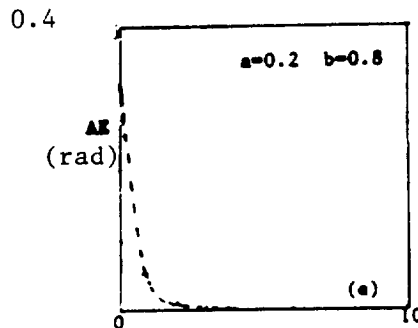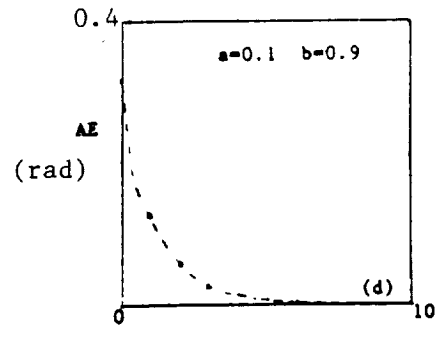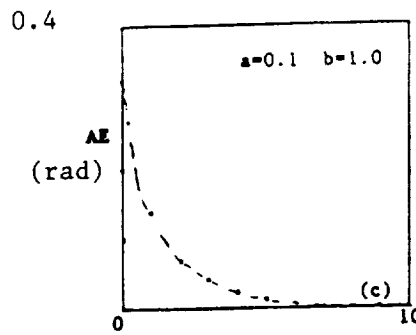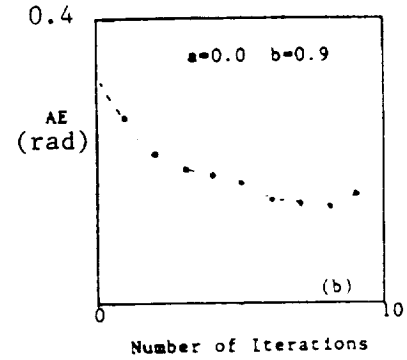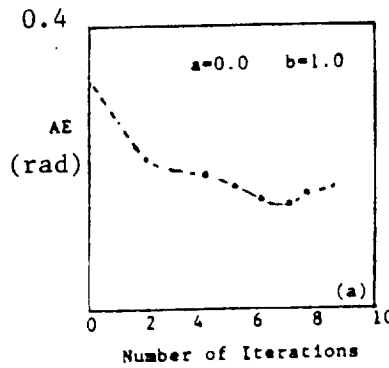
$X_c = \sin \omega t$

$\omega = 2 \pi \, rad/sec$

Fig. 6. Error Convergence of the Servosystem with the Learning Control Scheme
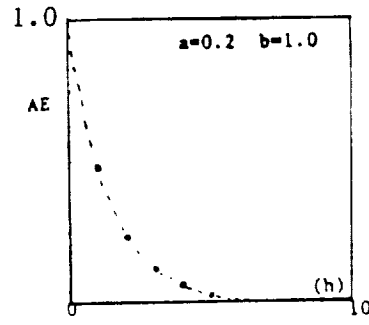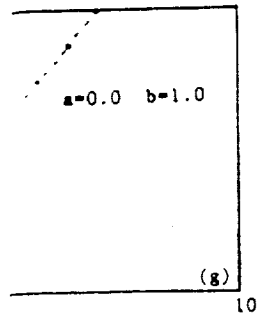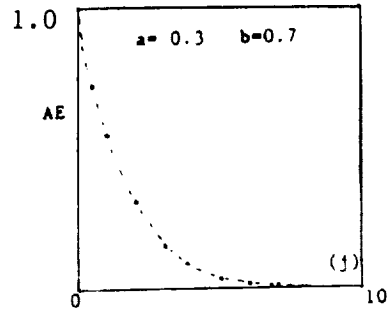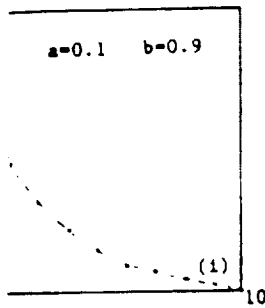
15

a=0.0 b=1.0

(g)

a=0.2 b=1.0

AE

(h)

aber of Iterations

a=0.1 b=0.9

(i)

a= 0.3 b=0.7

AE

(j)

$_c$ = sin (wt)   w = 4 πrad/sec

ror Convergence of the Servosystem with the Learning
ıtrol Scheme

Actual Trajectory

Average Error = 0.0035

Time (secs.)
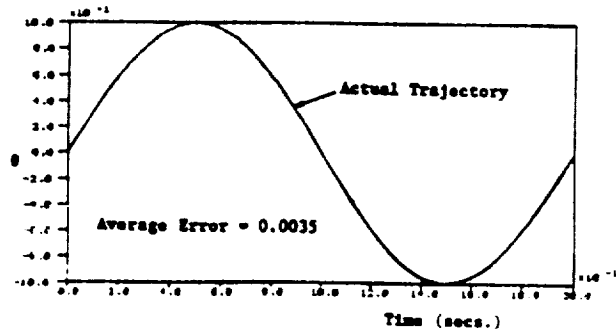
ıse of the System for the Modified Command Input

16
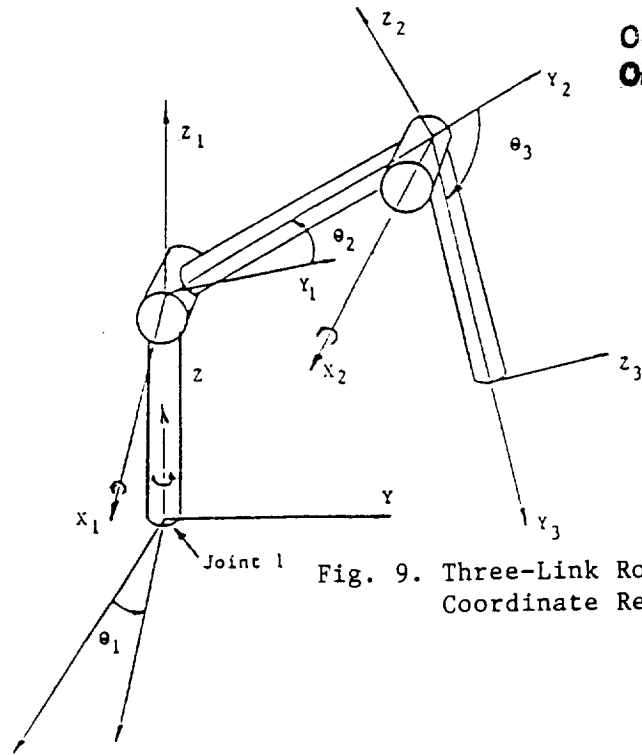
Fig. 9. Three-Link Robot Arm and the Coordinate Reference Frames
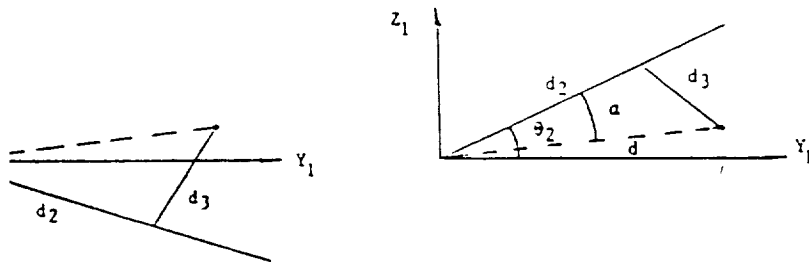
ible Configurations of Links 2 and 3 to reach a
int

r of Iterations

Number of Iterations
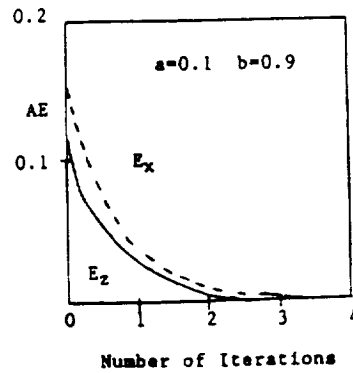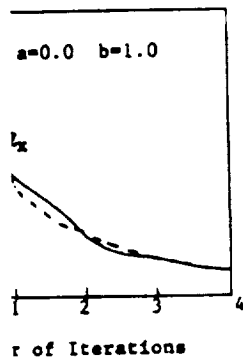
Convergence for a Three-Link Robot Computer Model
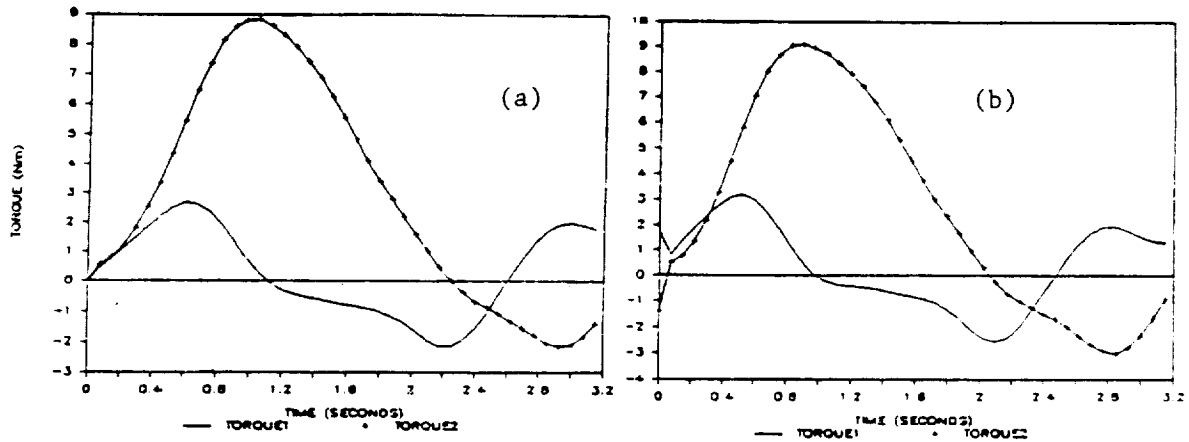
Fig. 12. Torque Characteristics (a) First Trial
                                  (b) After Four Iterations

Z (m)

X(m)
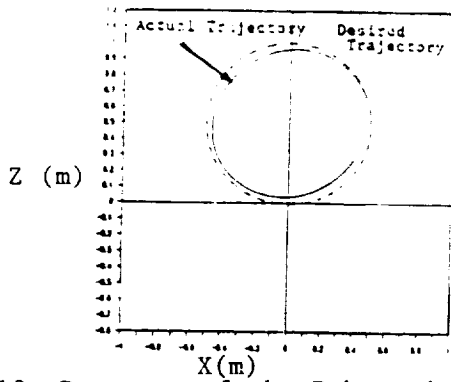
Fig. 13. Response of the Robot with
          Command Input Same as
          Desired Trajectory

X(m)
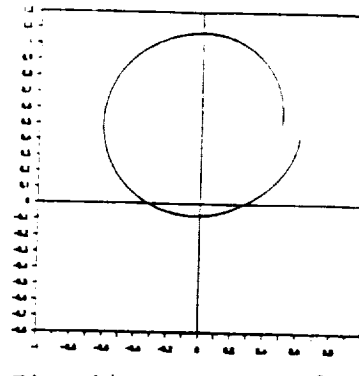
Fig. 14. Response of the Robot
          After 1 Iteration

Z(m)
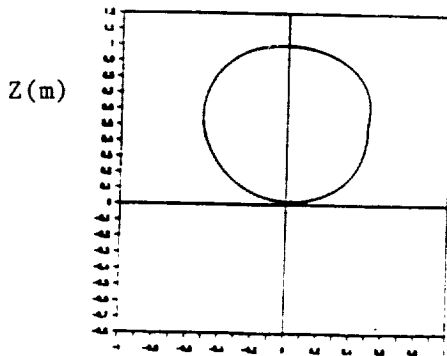
Fig. 15. Response of the Robot
          After Four Iterations
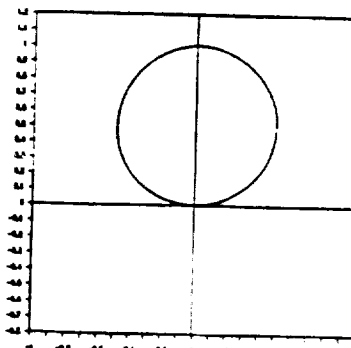
Fig. 16. Response of the Robot for
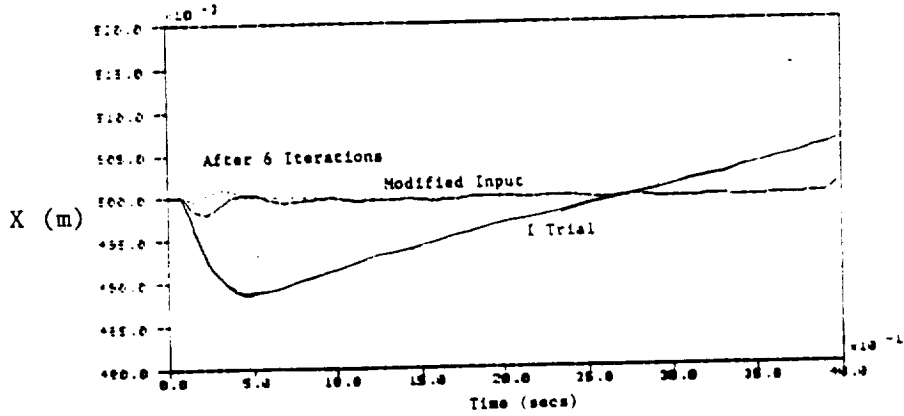   a Modified Command Input

18

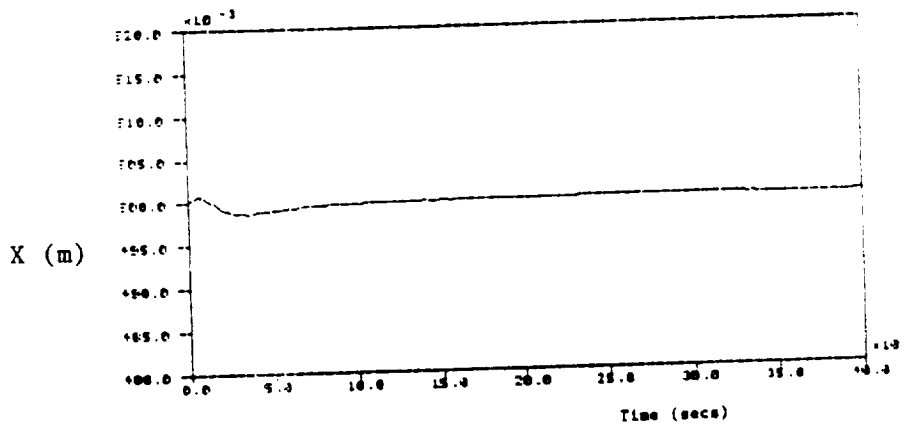**Fig. 17.** Response of the Robot After 6 Iterations and A Modified Command Input

Fig. 18. Response of the Robot for a Modified Command Input (t_p = 0.2 secs.)

Waist (Joint 1)   Shoulder (Joint 2)

Elbow (Joint 3)

Wrist Rotation (Joint 4)
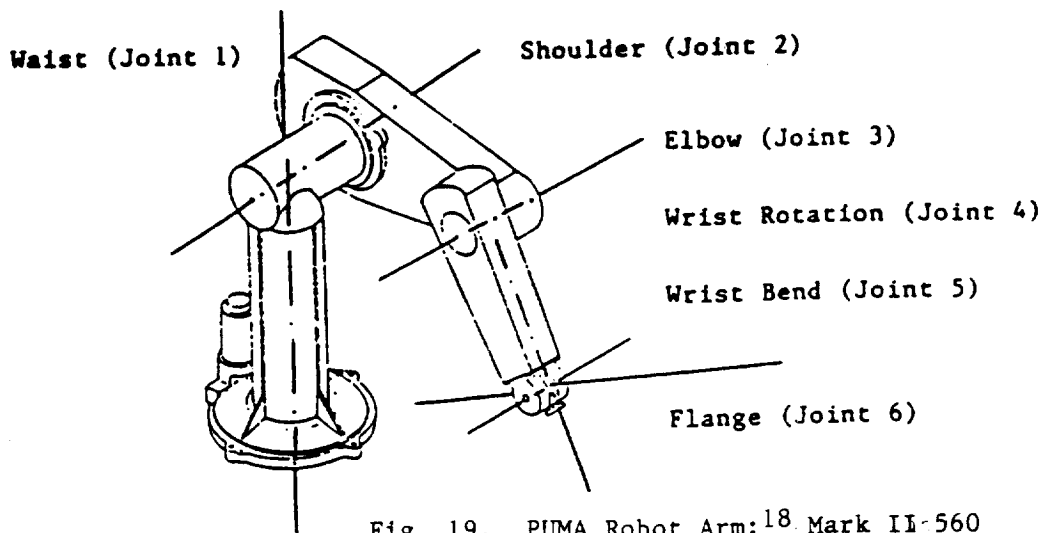
Wrist Bend (Joint 5)

Flange (Joint 6)

Fig. 19. PUMA Robot Arm:[18] Mark II-560

Fig. 20 STRAIGHT LINE ALONG Y AXIS,
ATTACHED PAYLOAD NONE, SPEED 150
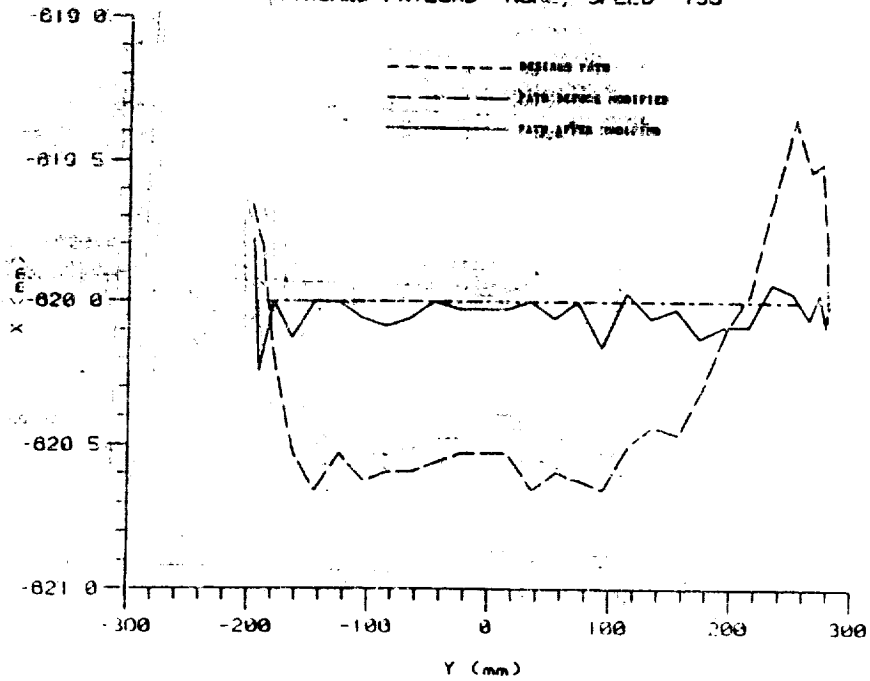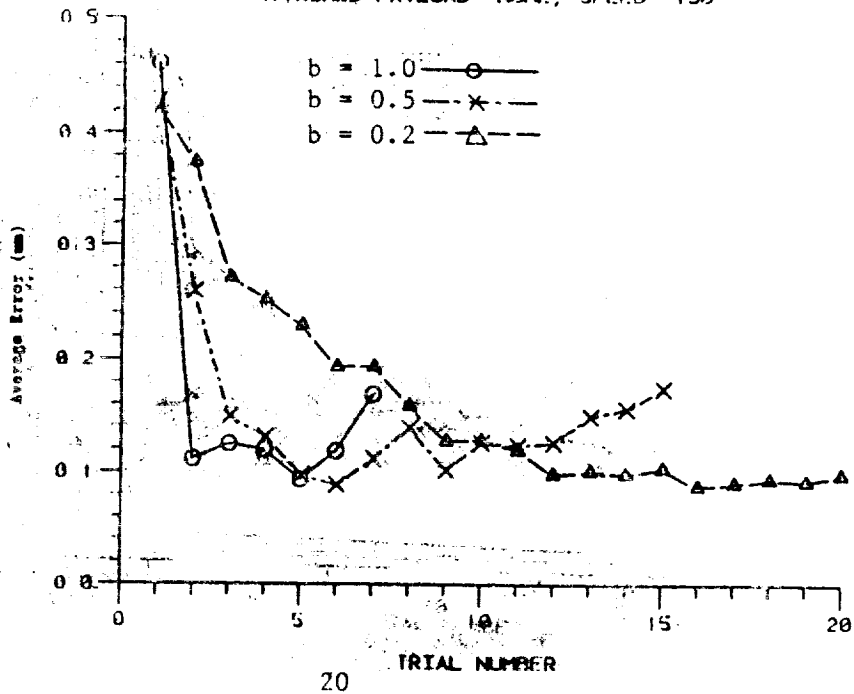

Fig. 21 STRAIGHT LINE ALONG Y AXIS,
ATTACHED PAYLOAD NONE, SPEED 150

20