Position and Force Hybrid Control of Robotic Manipulator by Neural Network (Adaptive Control of 2 D.O.F. Manipulators)

Masatoshi TOKITA *, Toyokazu MITUOKA*,
Toshi FUKUDA** ,Takanori SHIBATA** and Fumihito ARAI**

*Kisarazu National College of Technology
Kiyomidai-higashi 2-11-1, Kisarazu-shi, Chiba, 292, Japan
**Nagoya University, Department of Mechanical Engineering,
Furo-cho 1, Chikusa-ku, Nagoya, 464-01, Japan

Abstract In this paper, a position/force hybrid control of a robotic manipulator based on a neural network model is proposed with consideration of the dynamics of objects and the orientations of the robotic manipulator. This proposed system consists of standard PID controller, the gains of which are augmented and adjusted depending on objects and orientations of manipulators through a process of learning.

The authors proposed a similar method previously for the force control of one degree-of-freedom manipulator. The proposed method shows the better performance than the conventional PID controller, yielding the wider range of applications. This paper shows the similar structure of the controller via the neural network model applicable to the cases of position/force hybrid control of multi degrees-of-freedom manipulators. Simulations as well as experiments are carried out for the case of the two degree-of-freedom robotic manipulators. The results show the applicability and adaptability of the proposed method to the position/force hybrid control of manipulators.

1. Introduction

Conducting high level operations such as insertion and grinding with a robot manipulator requires control not only of manipulator position but also of the manipulator and object's mutual force. However, with a control system involving interaction with an object, dynamic characteristics of the object are contained in the feedback loop of the system, which causes the dynamic characteristics of the object to change, thereby deteriorating control performance.

In the previous reports[1][2], the authors proposed force control methods of one degree-of-freedom manipulator using a neural network model with unknown objects. The proposed method shows the better performance than the conventional PID type of controller, yielding to the wider range of applications, consequently. This paper shows the similar structure of the controller via the neural network model applicable to the cases of the position/force hybrid control of multi degree-of-freedom manipulators. There are many researches on the hybrid control schemes of multi degree-of-freedom manipulators[3]-[6]. The multi degree-of-freedom manipulator is nonlinear system. As to the nonlinear system, Khatib[4] shows nonlinear dynamic decoupling approach. Stability analysis of the hybrid control is presented by Yabuta[5]. However, the method to be taken is not yet decided upon.

Simulations and experiments are carried out for the case of two degree-of-freedom manipulators. The ability of the proposed controller is shown by the robustness against the dynamics of an object and orientation attitude of the manipulators. The result show the applicability and adaptability of the proposed controller to the position/force hybrid control of robotic manipulators.

2. Construction of Hybrid Control System

2-1. Control Object and Hybrid Control System

This paper deals with a position/force hybrid control system of a multi degree-of-freedom manipulator, shown in Fig.1. It detects a contact force with a force sensor utilizing a strain gauge installed on

CH 3065-0/91/0000-0113 $1.00 © IEEE

the wrist and controls a contact force between an object gripped by hand and the fixed object.

Figure 2 shows the construction of a hybrid control system using a neural network model. This hybrid control system uses the method proposed by Raibert and Craig[7]. In this system, the neural network model dose not control the robotic manipulator directly. It estimates the dynamics of an object and changes feedback gain of PID controller accordingly to prevent control performance from deteriorating.

The joint driving force is given by

$$\tau = \tau_p + \tau_f \quad (1)$$

$$\tau_p = K_{pp}q_e(t) + K_{pi}\int q_e(t)dt + K_{pd}\dot{q}_e(t) \quad (2)$$

$$\tau_f = K_{fp}t_e(t) + K_{fi}\int t_e(t)dt \quad (3)$$

$$q_e(t) = J^{-1}(I-S)\{X_d(t) - \Lambda(q(t))\} \quad (4)$$

$$\dot{q}_e(t) = J^{-1}(I-S)\{\dot{X}_d(t) - J\dot{q}(t)\} \quad (5)$$

$$\tau_e(t) = J^T S\{F_d(t) - TF(t)\} \quad (6)$$

where

K_{pp}, K_{pi}, K_{pd} : position feedback gain,

K_{fp}, K_{fi} : force feedback gain,

J : Jacobian matrix,

S : compliance selection matrix,

L : kinematic function of manipulator,

T : force transformation, t : time,

X : position vector, F : force vector,

q : manipulator position in joint coordinates.

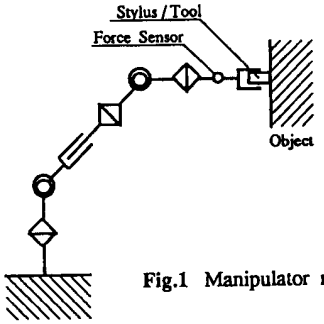


Fig.1 Manipulator model

2-2. Neural Network Model

The neural network model uses PDP model with the manipulator's position, the end-effector position in the constraint coordinate, the contact force with an object, and a force desired value as input signals and each gain value of PID control as output signals. The states of the units in each layer are based on the following equations:

$$u_i = \Sigma_j W_{ij} f(u_j) \quad (7)$$

$$f(u_i) = \begin{cases} 1/\{1+\exp(-u_i)\} & \text{(input and intermediate layer)} \\ u_i & \text{(output layer)} \end{cases} \quad (8)$$

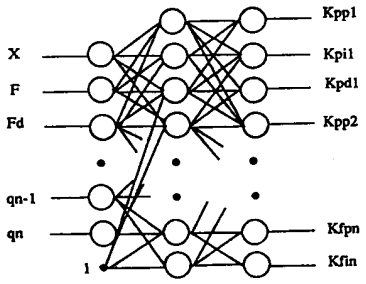


Fig.3 Neural network model

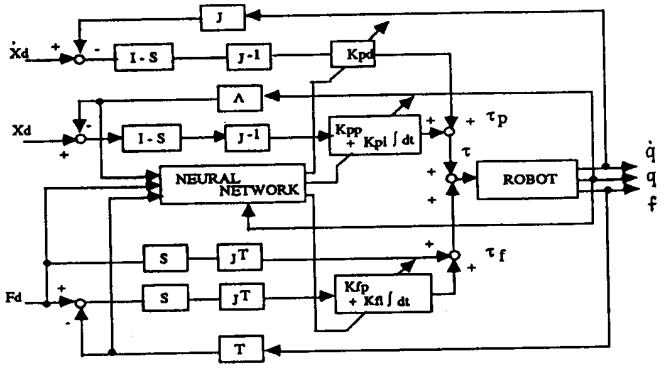


Fig.2 Control system diagram

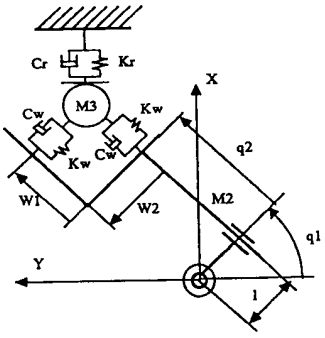


Fig.4 Mathematical model for a manipulator

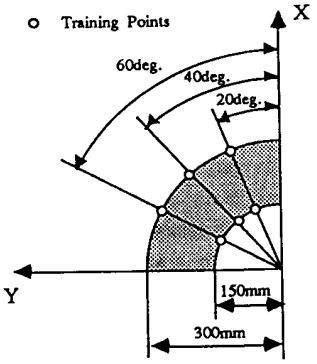


Fig.5 Training points

where u_i and W_{ij} represent the activity of unit i and connection strength from units j to i, respectively. Bias(connection strength from a unit constantly with a value of 1) was applied to each unit of the intermediate layer.

For learning of the neural network model, the following error back propagation algorithm[8] was used and each feedback gain experimentally found with known objects was used for teaching signals:

$$\Delta W(k+1) = -\eta \partial E/\partial W + \alpha \Delta W(k) \qquad (9)$$

where W(k), E, and η and α represent correction amount for connection, output square error and constants, respectively. Learning was conducted with every unit of the three layers with weight for connection varied for each sampling time.

3. Simulation Method and Results

3-1. Mathematical Model of Control Object and Control System

In order to examine the proposed neuromorhpic hybrid control, we selected the two degree-of-freedom manipulator shown in fig.4. Joints of the manipulator were controlled in a plane whose normal aligns with the gravity vector. The constraint surface was chosen to lie in this plane with the X direction. The X-axis is force controlled, and Y-axis is position controlled. The dynamic equations of the manipulator are as follows:

$$\ddot{q}_1 = (\tau_1 - C_1\dot{q}_1 + q_2K_w w_2 + lK_w w_1)/I_1 \qquad (10)$$

$$\ddot{q}_2 = (\tau_2 - C_2\dot{q}_2 + K_w w_1)/(M_2 + M_3) \qquad (11)$$

$$\dot{w}_1 = \{-K_w w_1 - C_w \dot{w}_1 - f_x \cos(q_1)\}/M_3 \qquad (12)$$

$$\dot{w}_2 = \{-K_w w_2 - C_w \dot{w}_2 + f_x \sin(q_1)\}/M_3 \qquad (13)$$

$$f_x = K_r X_c + C_r \dot{X}_c \qquad (14)$$

$$X_c = (q_2 + w_1)\cos(q_1) + (l - w_2)\sin(q_1) - X_r \qquad (15)$$

q_1, q_2 : manipulator position in joint coordinates,
w_1, w_2 : wrist force sensor deflections,
τ_1, τ_2 : joint driving torque and force,
C_1, C_2 : viscosity coefficients,
I_1 : moment of inertia at joint 1,
C_w : viscosity coefficient of force sensor and manipulator,
K_w : spring constant of force sensor and manipulator,
M_2, M_3 : mass of manipulator links,
l : link offset, fx : reaction force,
K_r : spring constant of an object,
C_r : viscosity coefficient of an object,
X_c : location of an object.

Table 1 shows numerical values of parameter used in simulations.

The neural network model comprised three layers with five units for the input layer and ten units for each intermediate and output layer, while η and α, learning constants in expression (9), were set to 0.01 and 0.3, respectively. Figure 5 shows training points in the removable area of this manipulator.

3-2. Simulation Results of Set-Point-Control

Simulations were conducted with respect to the fixed feedback gain method and the adaptive gain method using the neural network model. A spring constant of 8×10^2(N/m) was used for soft objects and that of 6×10^4(N/m) for hard ones. Figure 6 shows results by fixed gain method. This figure shows reaction

force(F), displacement of the object(X-X0) and position deviation(Y-Yd). As shown here, faulty selection of gain makes the controller unadaptable to changes in the dynamics of the object, thereby causing an oscillation phenomenon. Figure 7 shows results by the adaptive gain method using the neural network model. It was found that this method enables the controller to adapt to changes in the dynamics of the object to improve positively.

We have simulated how the neural network can recognize unknown objects and respond to them. Table 2 shows the squared errors of the controllers faced to various objects. The learning points are only two, i.e. $Kr=8\times10^2$(N/m) and 6×10^4(N/m). When the stiffness of an object increases the squared error by the fixed gain controller increases drastically, because the fixed gain controller produces oscillation in response. But the adaptive gain does not produce such large distinction in the squared error. These data indicate that this adaptive method can ensure a stable response over the entire range between two learning points of the object's stiffness.

Figures 8 and 9 show the results of set-point-control at various orientations of manipulator. Figure 8 shows the results by fixed gain controller. As shown here, the manipulator is unstable to the changes of orientation. Figure 9 shows the results by adaptive gain controller. It is evident that this adaptive method is robust to the changes of manipulator orientation.

3-3. Simulation Results of Contouring Motion Control

Figures 10 and 11 show simulation results of contouring motion control. Figure 10 shows results with respect to changes in the dynamics of objects using fixed gain method. It shows that the fixed gain controller is unstable to changes in the dynamics of the object. Figure 11 shows results using the adaptive gain method. The improvement due to this proposed method is evident.

4. Experimental Method and Results

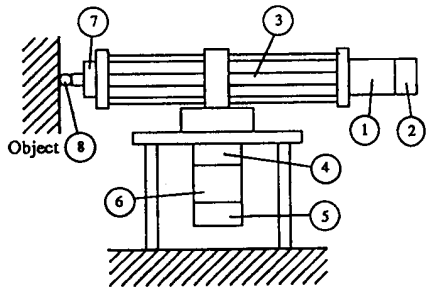
4-1. Experimental System and Experimental Method

In order to examine the effectiveness of the proposed method, we conducted some experiments. The overview of the manipulator is shown in Fig.12 and Fig.13. The system uses D.C. servo motors as actuators. The encoder mounted on the motor shaft detects position and force by force sensor(strain gauge) mounted on the manipulator tip. The controller is implemented in personal computer (CPU 8086 and FPU 8078), and sampling time for control is 20ms.

With the size of objects to handle unknown, recognition movement was first conducted. Recognition movement involved the manipulator approaching the object at constant speed while controlling speed and recognizing contact with the object where the force sensor output exceeds the set value(0.2N). From that point, the manipulator was controlled by fixed feedback gain and adaptive feedback gain.

4-2. Experimental Results

Figures 14 and 15 show results of set-point-control using the experimental system in Fig.12. The figures all show responses of the system after recognition movement. A soft tennis ball(64mm in dia.: the value of spring constant,2×10^2N/m) was used as a soft object, and steel plate(the value of spring constant,2×10^{11}N/m) as a hard one. Figure 14 shows results of set-point-control using the fixed gain. Pushing a hard object with a soft object using feedback gain capable of responding as in this figure causes an oscillation phenomenon, resulting in deterioration in control performance. As in the simulation, it has also been made clear in an experiment that the adaptive range is limited with respect to changes in the dynamics of objects. Figure 15 shows results of set-point-control using the adaptive gain method proposed in this study. It has been found that stable control movement can be



1. DC Motor 2. Encoder 3. Screw
4. Gear 5. Encoder 6. DC Motor
7. Force Sensor 8. Ball Bearing

Fig.12 Experimental system

achieved with both soft and hard objects and that adaptability can be expanded with the changes of object's dynamics.

Figures 16 and 17 show the results of set-point-control at various orientations of manipulator. Figure 16 shows the results by the fixed gain controller. As shown here, the manipulator is unstable to the changes of orientation. On the other hand, the adaptive gain controlled manipulator is stable to the changes of orientation, shown in Fig.17.

The squared errors for force control at various orientations of manipulator are shown in Fig.18. These data indicate that this adaptive method can ensure a stable response besides learning points (shown in fig.5).

5. Conclusion

In this paper, we have proposed an adaptive hybrid control method which utilizes a neural network model, and performed simulations and experiments for a two degree-of-freedom manipulator system. The proposed method was shown a wider applicability with respect to the following things:

(1) changes of the object's dynamics

(2) changes of the manipulator's orientation

reference

(1)Tokita, Mituoka, Fukuda and Kurihara, "Force Control of Robotic Manipulator by neural network", JRSJ, vol.7,No.1, pp.47-51, 1989

(2)Tokita, Mituoka, Fukuda and Kurihara, "Force Control of Robotic Manipulator by Neural Network (Experimental Results and their Evaluation of One Degree-of-Freedom Manipulator)", JRSJ, vol.8, No.3, pp.292-299, 1990.

(3)C.H.An, J.M.Hollerbach, "Dynamic Stability Issues in Force Control of Manipulators", Proc. of IEEE International Conference on Robotics and Automation, pp.890-896, 1987.

(4)Khatib, "A unified Approach for Motion and Force Control of Robot Manipulators (The Operational Space Formulation)", IEEE Journal of Robotics and Automation, Vol.RA-3, No.1, pp.43-53, 1987.

(5)Yabuta, "Stability Feature of the Hybrid Position/ Force Control Scheme for Robot Manipulator", Proc. of IEEE Workshop on Intelligent Robot and System, pp.121-125, 1988.

(6)Fukuda, Kitamura, and Tanie, "Adaptive force control of manipulators with consideration of object dynamics", Proc. of IEEE International Conference on Robotics and Automation, pp.1543-1548, 1987.

(7)Raibert and Craig, "Hybrid Position/Force Control of Manipulators", ASME, Journal of Dynamic Systems, Measurement and Control 102, pp.126-133, 1981.

(8)Rumelhart, Hinton and Williams, "Leaning representations by backpropagation error", Nature, 323, pp.533-536, 1986.

(9)Fukuda, Shibata, Tokita and Mituoka, "Neural Servo Controller (Adaptation and Learning)",Proc. of IEEE International Workshop on Advanced Motion Control, pp.107-115, 1990.

Table 1 Manipulator parameters used for simulation

M 2	3	K g
M 3	0 . 1	K g
C 1	0 . 1 1	N m s
C 2	2 2	N s / m
l	0 . 0 5	m
K w	3 × 1 0 ⁴	N / m
C w	3 . 7 5	N s / m
I 1	$0.0324 \cdot \phi$ $2(q2-0.18)^2$	K g m ²
K r (soft)	8 × 1 0 ²	N / m
C r (soft)	1 1 . 1	N s / m
K r (hard)	5 × 1 0 ⁴	N / m
C r (hard)	1 0 . 5	N s / m

Table 2 Squared error

Kr (N/m)	Fixed Gains		Adaptive Gains	
	force	position	force	position
800	1.000	1.000	1.011	0.805
15000	0.621	0.043	0.517	0.013
30000	0.474	0.016	0.435	0.005
50000	5.015	3.573	0.566	0.005
60000	49.916	14.989	1.267	0.068

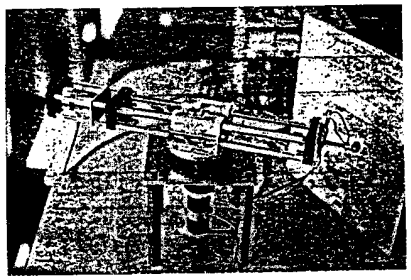
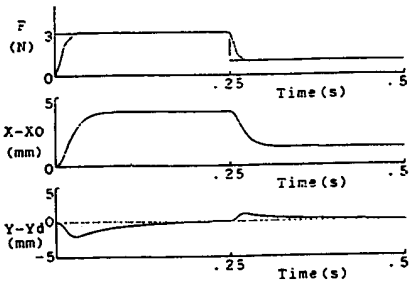
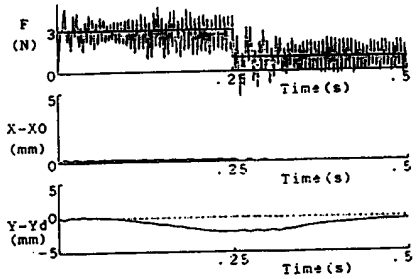


Fig. 13 Experimental system

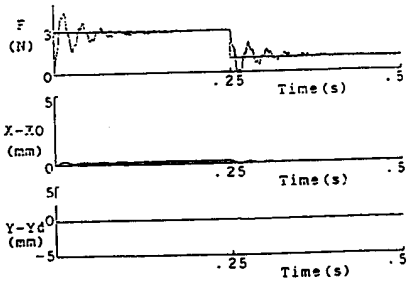


(a) soft object

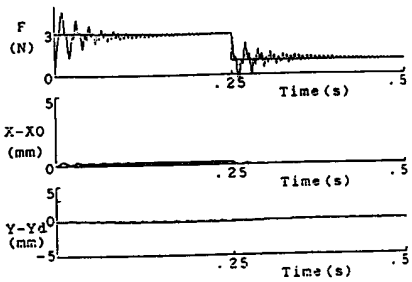


(b) hard object

Fig. 6 Simulation results with the fixed gain control (influence of the dynamics of objects)

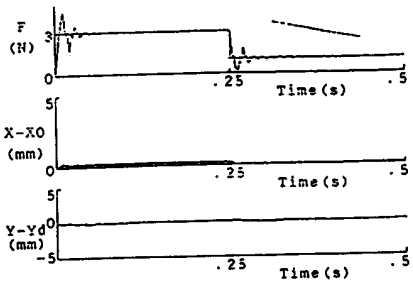


(a) q1=45° q2=150mm

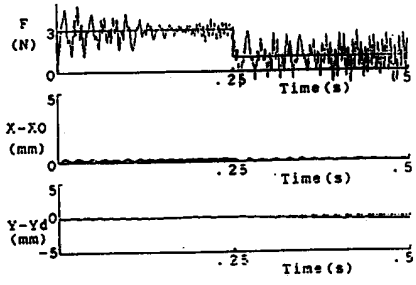


(b) q1=45° q2=300mm

Fig. 7 Simulation results with the adaptive gain control (influence of the dynamics of objects)

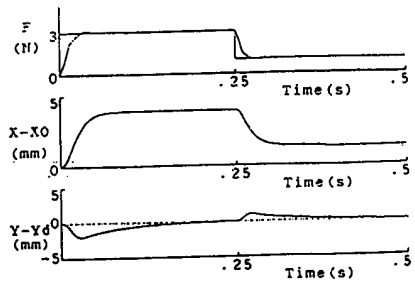


(a) q1=45° q2=150mm

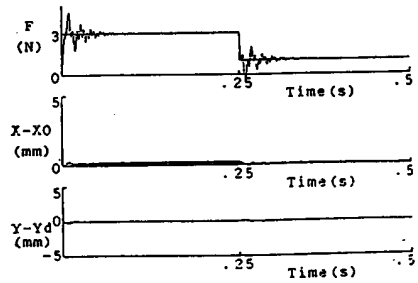


(b) q1=45° q2=300mm

Fig. 8 Simulation results with the fixed gain control (influence of the orientations)

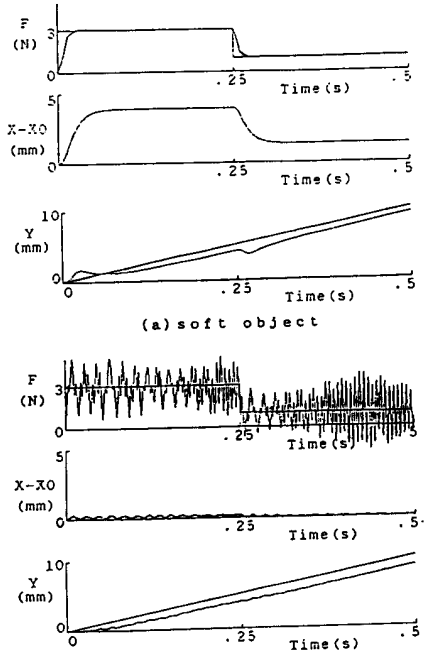


(a) soft object



(b) hard object

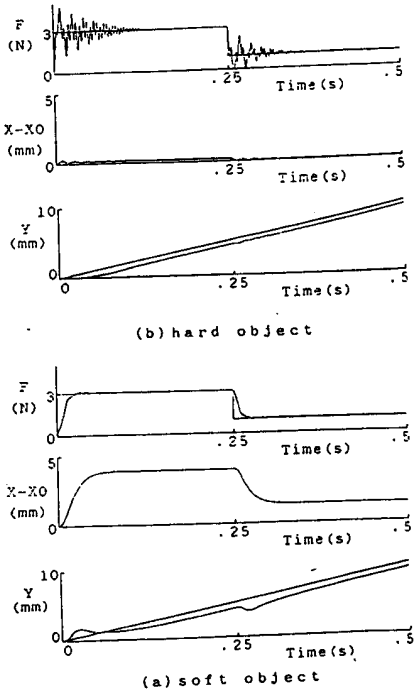
Fig. 9 Simulation results with the adaptive gain control (influence of the orientations)



(a) soft object

(b) hard object

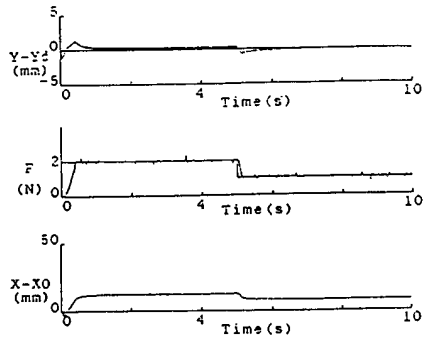
Fig. 10 Simulation results with the fixed gain control (contouring motion control)



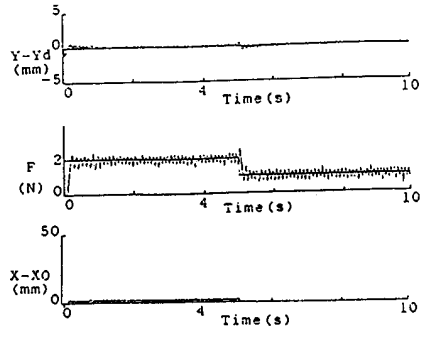
(b) hard object

(a) soft object

Fig. 11 Simulation results with the adaptive gain control (contouring motion control)

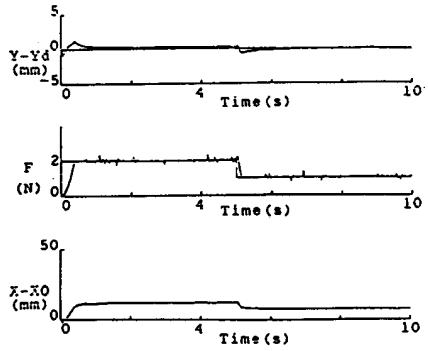


(a) soft object

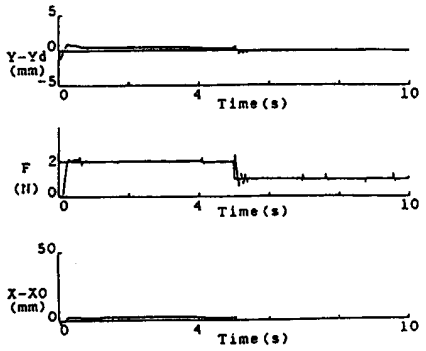


(b) hard object

Fig. 14 Experimental results with the fixed gain control (influence of the dynamics of objects)

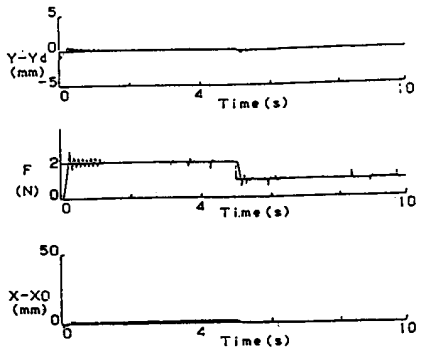


(a) soft object

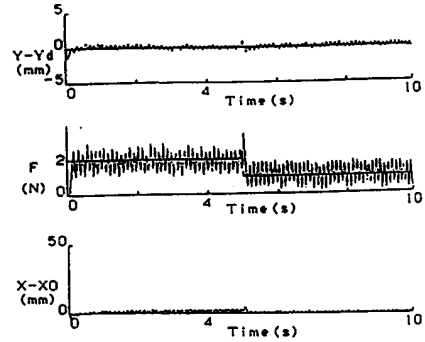


(b) hard object

Fig. 15 Experimental results with the adaptive gain control (influence of the dynamics of objects)

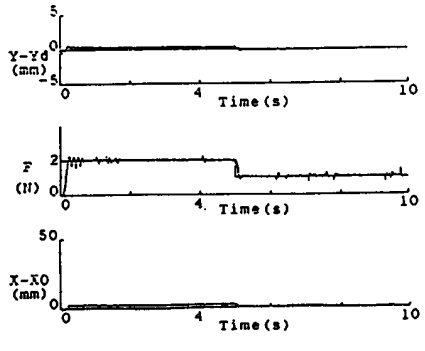


(a) q1=40° q2=150mm

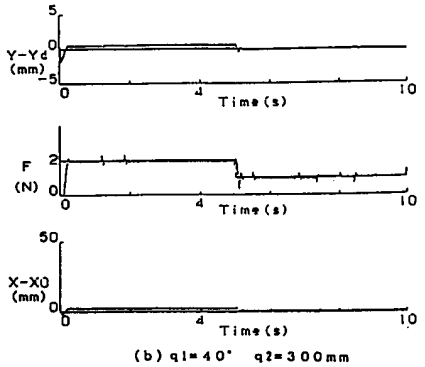


(b) q1=40° q2=300mm

Fig. 16 Experimental results with the fixed gain control (influence of the orientations)



(a) q1=40° q2=150mm



(b) q1=40° q2=300mm

Fig. 17 Experimental results with the adaptive gain control (influence of the orientations)

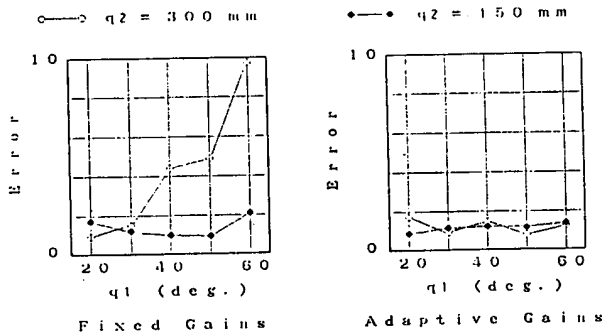


Fig. 18 Squared error for force control

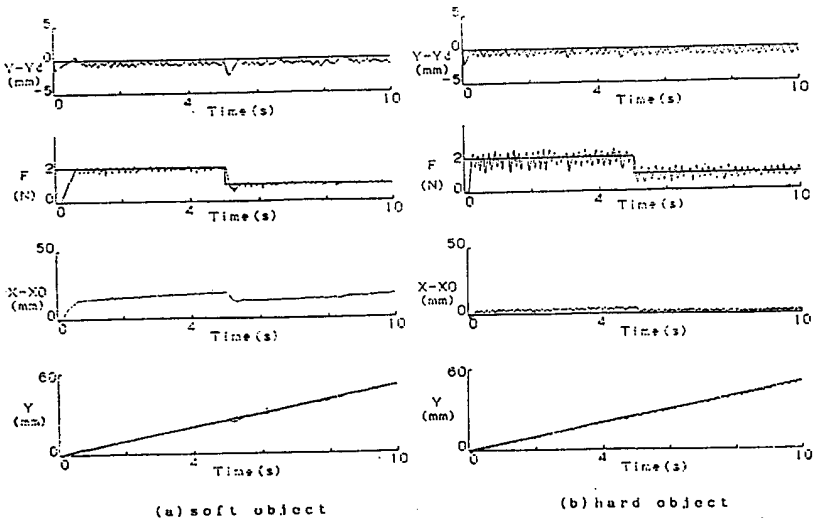


Fig. 19 Experimental results with the fixed gain control (contouring motion control)

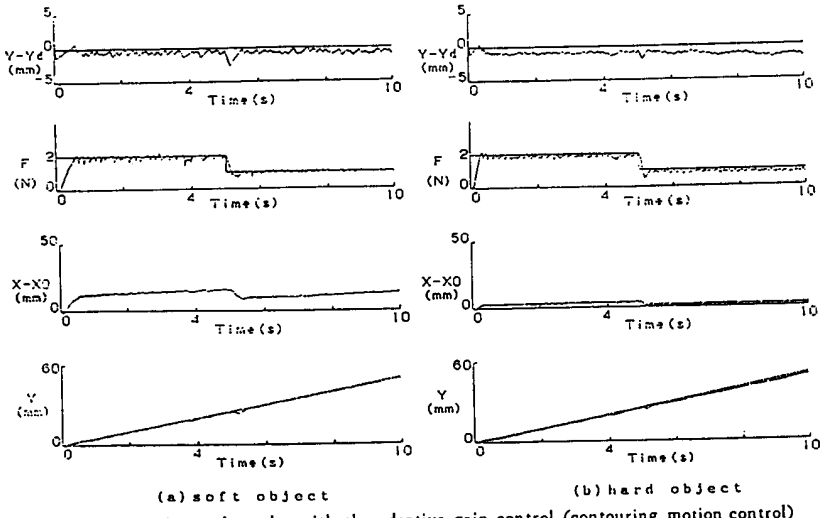


Fig. 20 Experimental results with the adaptive gain control (contouring motion control)