Optimal Trajectory Control for Robot Manipulators using Evolution Strategy and Fuzzy Logic

Jin-Hyun Park, Hyun-Sik Kim, and Young-Kiu Choi

Abstract : Like the usual systems, the industrial robot manipulator has some constraints for motion. Usually we hope that the manipulators move fast to accomplish the given task. The problem can be formulated as the time-optimal control problem under the constraints such as the limits of velocity, acceleration and jerk. But it is very difficult to obtain the exact solution of the time-optimal control problem. This paper solves this problem in two steps. In the first step, we find the minimum time trajectories by optimizing cubic polynomial joint trajectories under the physical constraints using the modified evolution strategy. In the second step, the controller is optimized for robot manipulator to track precisely the optimized trajectory found in the previous step. Experimental results for SCARA type manipulator show that the proposed method is very useful.

Keywords : optimal trajectory planning, MIMO fuzzy controller, modified evolution strategy

I. Introduction

Robot manipulators have become to be indispensable for factory automation. In order to increase the productivity, it is desirable to move the manipulators as fast as possible. However, the manipulators have some physical constraints such as the limits of position, velocity, acceleration, and jerk. Therefore, It is desirable to solve the minimum-time control problem [1][5]. But it is very difficult to solve the problem. In this paper, we solve this problem in two steps. In the first step, we find the optimal trajectory along which the robot moves under the constraints. In the second step, the fuzzy controller is optimized for the robot to precisely track the trajectory optimized in the previous step.

In the first step, the path of robot is specified by a sequence of knot points in the task space. The trajectory between the knot points is described by the cubic polynomials, and parameterized by time intervals between the knot points. Each time interval should be optimized so that the total traveling time, which can be regarded as a performance index, may be minimized under the constraints. To find the optimal intervals, the flexible polyhedron search method was used by Lin et al. [1][5]. But this algorithm has characteristics of local search whose performance depends on initial conditions [2]. In this paper, the time intervals are optimized by using the modified Evolution Strategy (ES) [3] which is a kind of evolutionary algorithms and is known to be simple and have excellent global search feature.

Manuscript received : Oct. 21, 1997., Accepted : Nov. 26, 1998.
Jin Hyun Park : Department of Industrial Automation Engineering, Chinju National University.
Hyun Sik Kim : Agency for Defense Development.
Young Kiu Choi : Department of Electrical Engineering, Pusan National University.
※ This work was supported by Pusan National University.

In the second step, using the modified ES we conduct tuning operations for a fuzzy logic controller of the manipulators to precisely track the trajectories optimized in the first step. In this case, the performance index can be formulated as an integral of squared error between the desired trajectory and actual trajectory. A conventional fuzzy logic method can not be easily applied to the manipulators which are Multi-Input-Multi-Output (MIMO) systems, and thus requires a large number of rules in the rule base. Recently, a New Fuzzy Reasoning Method (NFRM) was proposed and turned out to be simple and superior to the Zadeh's fuzzy reasoning method [4]. This fuzzy reasoning method is composed of fuzzification, a relation matrix and defuzzification. The fuzzy reasoning method uses the relation matrix in place of the general fuzzy reasoning rules. Therefore, fuzzy reasoning method does not require long calculation time (reasoning time) and its reasoning accuracy is good in comparison with that of the general fuzzy reasoning rules and its structure is simple. To handle the MIMO system, we extend the NFRM by enlarging the number of relation matrix elements in fuzzy reasoning method. However, it is difficult to choose a proper relation matrix of the NFRM and the extended fuzzy reasoning method. Therefore, we do efficiently the tuning operation for the extended fuzzy reasoning method using the modified ES. Finally, we have applied the trajectory planning and control method to the SCARA-type robot manipulator, and found that the control performance is very effective in the sense of fast trajectory planning and accurate trajectory tracking.

II. Formulation of robot manipulator trajectory

In this paper, robot trajectory is constructed by a sequence of cubic polynomials, which was formulated by Lin et al. [1][5].

Since cubic polynomial trajectories are smooth and have small overshoot of angular displacement between two adjacent knot points, Lin et al. adopted the idea of using cubic spline polynomials to fit the segment between two adjacent knots. In order to satisfy the continuity conditions for the joint displacement, velocity, and acceleration on the entire trajectory for the cartesian path, two extra knot points with unspecified joint displacements must be added to provide enough degrees of freedom for solving the cubic polynomials under continuity conditions.

When the knot points in the joint space are specified by q_1, q_3, q_4, ..., q_{n-3}, q_{n-2}, q_n (q_2 and q_{n-1} are unspecified extra points.) and the time intervals d_i's between the knot points are given, the joint trajectory can be described as follows [1]. Let $Q_i(t)$, $\dot{Q}_i(t)$, $\ddot{Q}_i(t)$ and be position, velocity and acceleration between the knot points q_i and q_{i+1}, respectively.

Position :

$$Q_i(t) = \frac{\ddot{Q}_i(t_i)}{6d_i}(t_{i+1}-t)^3 + \frac{\ddot{Q}_i(t_{i+1})}{6d_i}(t-t_i)^3 + \left[\frac{q_{i+1}}{d_i} - \frac{d_i\ddot{Q}_i(t_{i+1})}{6}\right](t-t_i) + \left[\frac{q_i}{d_i} - \frac{d_i\ddot{Q}_i(t_i)}{6}\right](t_{i+1}-t_i) \quad (1)$$

where i = 1, 2, ... , n-1,

$$q_2 = q_1 + d_1 v_1 + \frac{d_1^2}{3}a_1 + \frac{d_1^2}{6}\ddot{Q}_1(t_2),$$

$$q_{n-1} = q_n + d_{n-1}v_n + \frac{d_{n-1}^2}{3}a_n + \frac{d_{n-1}^2}{6}\ddot{Q}_{n-2}(t_{n-1}).$$

Velocity :

$$\dot{Q}_i(t) = \frac{-\ddot{Q}_i(t_i)}{2d_i}(t_{i+1}-t)^2 + \frac{\ddot{Q}_i(t_{i+1})}{2d_i}(t-t_i)^2 + \left(\frac{q_{i+1}}{d_i} - \frac{d_i\ddot{Q}_i(t_{i+1})}{6}\right) - \left(\frac{q_i}{d_i} - \frac{d_i\ddot{Q}_i(t_i)}{6}\right) \quad (2)$$

where i = 1, 2, ... , n-1.

Acceleration :

$$\ddot{Q}_i(t) = \frac{-(t-t_{i+1})}{d_i}\ddot{Q}_i(t_i) + \frac{t-t_i}{d_i}\ddot{Q}_i(t_{i+1}) \quad (3)$$

where i = 1, 2, ..., n-1.

The accelerations described above in each time interval can be computed using the positions of knot points and the time intervals, given both velocities and accelerations at the initial and final knot points.

To obtain the minimum-time trajectory under the constraints, each time interval d_i has to be optimized. A flexible polyhedron search method was used to find the minimum-time trajectory [1]. However, the search

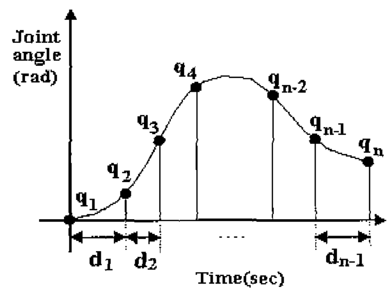


Fig. 1. Robot manipulator trajectory.

method is not only complicated but has characteristics of local search whose performance depends on the initial conditions.

In this paper, to find the optimal time intervals d_i's, we adopt the modified ES [3], which is simple and has excellent global search performance.

The mutation operation of the modified ES is described as follows. The offspring parameter vector X' is the sum of the parent parameter vector X and a Gaussian random vector with zero mean and standard deviation σ.

$$X' = X + N(0,\sigma)$$

$$\sigma = C/(f_B - f_A) \quad (4)$$

where C is a constant vector, f_B means the best fitness function, and f_A denotes the average fitness function.

A fitness function is necessary to find excellent individuals, and is defined as the inverse of the sum of traveling time and penalty functions caused by violating the given constraints. This means that a individual has a low evolutional probability in the case of long traveling time and a large penalty function value. The fitness function at the k-*th* individual has been chosen as follows:

$$f_k = \frac{1}{1 + \sum_{i=1}^{n-1} h_{ki} + p_1^2 + p_2^2 + p_3^2}$$

$$p_1 = W_{vel} \times \left(\sum_{j=1}^{N}\sum_{i=1}^{n-1} OVC(j,i)\right), \quad p_2 = W_{acl} \times \left(\sum_{j=1}^{N}\sum_{i=1}^{n-1} OAC(j,i)\right),$$

$$p_3 = W_{jerk} \times \left(\sum_{j=1}^{N}\sum_{i=1}^{n-1} OJC(j,i)\right) \quad (5)$$

where N and n represent the numbers of joints and knot points, and OVC(j,i), OAC(j,i) and OJC(j,i) denote penalties due to violation of constraints in the velocity, acceleration and jerk at the j-*th* joint in the time interval d_i. W_{vel}, W_{acl}, and W_{jerk} denote penalty weights.

The following equation shows the penalty function due to velocity constraints.

$$OVC(j,i) = \max_{t_i \le t \le t_{i+1}} \left\{ \left|\dot{Q}_i(t)\right| - VC_i \right\} \quad (6)$$

if OVC(j,i)<0 then OVC(j,i)=0, where VC_i is the velocity constraint.

Other penalty functions OAC(j,i) and OJC(j,i) can be defined in the same way.

III. MIMO fuzzy reasoning method

The fuzzy reasoning method by Z. Cao [4] is composed of fuzzification, a relation matrix and defuzzification. The fuzzy reasoning method uses the relation matrix in place of the general fuzzy reasoning rules. Our work will focus on the application of the fuzzy reasoning method to the MIMO system.

Considering the following form of fuzzy reasoning rules for a MIMO system,

If X_1 is A_{11}, ⋯, and X_n is A_{1n}
then Y_1 is B_{11}, ⋯, and Y_m is B_{1m}.
If X_1 is A_{21}, ⋯, and X_n is A_{2n}
then Y_1 is B_{21}, ⋯, and Y_m is B_{2m}.
⋯ ⋯ ⋯
If X_1 is A_{k1}, ⋯, and X_n is A_{kn}
then Y_1 is B_{k1}, ⋯, and Y_m is B_{km}.

where A_{1j}, A_{2j}, ⋯, A_{kj} are the linguistic terms of input variables X_j, and B_{11}, ⋯, B_{k1}, B_{12}, ⋯, B_{k2}, ⋯, B_{1m}, ⋯, B_{km} are the fuzzy subsets of the set of all linguistic terms of the outputs variables.

If inputs X_1, X_2, ⋯, X_n are a_1, a_2, ⋯, a_n, respectively, the results of fuzzy reasoning have the following form.

Let $X_1=a_1$, ⋯, and $X_n=a_n$.

Then Y_1 is b_1, ⋯, and Y_m is b_m. (7)

Meanwhile a_1, ⋯, a_n, b_1, ⋯, b_m are the real values of both input and output variables.

A relation matrix represents the relations between the linguistic descriptions of inputs X_1, X_2, ⋯, X_n and outputs Y_1, Y_2, ⋯, Y_m. Suppose that we have hi linguistic values C_{i1}, C_{i2}, ⋯, for the variable Y_i and the membership functions of B_{11}, ⋯, B_{1m}, B_{21}, ⋯, B_{2m}, ⋯, B_{k1}, ⋯, B_{km}.

B_{2m}, ⋯, B_{k1}, ⋯, B_{km}.

$$
\begin{aligned}
B_{11} &= b_{1,1}/C_{11} + b_{1,2}/C_{12} + \cdots + b_{1,h_1}/C_{1h_1} \\
&\vdots \\
B_{1m} &= b_{1,h_s-h_m+1}/C_{m1} + b_{1,h_s\ h_m+2}/C_{m2} + \cdots + b_{1,h_s}/C_{mh_m} \\
B_{21} &= b_{2,1}/C_{11} + b_{2,2}/C_{12} + \cdots + b_{2,h_1}/C_{1h_1} \\
&\vdots \\
B_{2m} &= b_{2,h_s-h_m+1}/C_{m1} + b_{2,h_s-h_m+2}/C_{m2} + \cdots + b_{2,h_s}/C_{mh_m} \\
&\cdots\cdots \\
B_{k1} &= b_{k,1}/C_{11} + b_{k,2}/C_{12} + \cdots + b_{k,h_1}/C_{1h_1} \\
&\vdots \\
B_{km} &= b_{k,h_s-h_m+1}/C_{m1} + b_{k,h_s-h_m+2}/C_{m2} + \cdots + b_{k,h_s}/C_{mh_n}
\end{aligned} \tag{8}
$$

where $h_s = h_1 + h_2 + \cdots + hm$, and $b_{i,j}$ represents the j-*th* membership degree of the output varable Y_i.

The relation matrix is shown as follows:

$$
\begin{array}{ccccccccc}
 & Y_1 & & & \cdots & & Y_m & & \\
C_{11} & C_{12} & \cdots & C_{1h_1} & \cdots & C_{m1} & C_{m2} & \cdots & C_{mh_m}
\end{array}
$$

$$
\begin{pmatrix}
b_{1,1} & b_{1,2} & \cdots & b_{1,h_1} & \cdots & b_{1,h_s-h_m+1} & b_{1,h_s\ h_m+2} & \cdots & b_{1,h_s} \\
b_{2,1} & b_{2,2} & \cdots & b_{2,h_1} & \cdots & b_{2,h_s-h_m+1} & b_{2,h_s-h_m+2} & \cdots & b_{2,h_s} \\
\vdots & & \vdots & \vdots & \cdots & \vdots & & \vdots & \vdots \\
b_{k,1} & b_{k,2} & \cdots & b_{k,h_1} & \cdots & b_{k,h_s-h_m+1} & b_{k,h_s-h_m+2} & \cdots & b_{k,h_s}
\end{pmatrix}
$$

For the input X_i, suppose that there are ki possible linguistic descriptions denoted by {X(i,1), X(i,2), X(i,3), ⋯, X(i,ki)}. Then, for the real value $X_i = a_i$, we can find its corresponding fuzzy subset as follows:

$$\tilde{a}_i = a(i,1)/X(i,1) + a(i,2)/X(i,2) + \cdots + a(i,k_i)/X(i,k_i) \tag{9}$$

where a(i, j) represents the membership degree to which the real value ai belongs to the linguistic description X(i, j) for j=1, 2, ⋯, k_i.

Thus, $X_1 = a_1$ belongs to the linguistic descriptions X(1,1), X(1,2), ⋯, X(1,k_1) with membership degrees a(1,1), a(1,2), ⋯, a(1,k_1), respectively. $X_2 = a_2$ belongs to the linguistic descriptions X(2,1), X(2,2), ⋯, X(2,k_2) with the membership degrees a(2,1), a(2,2), ⋯, a(2,k_2), respectively. $X_n = a_n$ belongs to the linguistic descriptions, X(n,1), X(n,2), ⋯, X(n,kn) with the membership degrees a(n,1), a(n,2), ⋯, a(n,kn), respectively. Thus, the group of real values ($X_1 = a_1$, $X_2 = a_2$, ⋯, $X_n = a_n$) belongs to $k(=k_1 \times k_2 \times \cdots \times k_n)$ groups of linguistic descriptions with different degrees. Therefore, X(1,1), X(2,1), ⋯, X(n,1) with degrees {a(1,1), a(2,1), ⋯, a(n,1)}, will be in the set of {(A_{i1}, A_{i2}, ⋯, A_{in})} (i = 1, 2, ⋯, k), where {a(1,1), a(2,1), ⋯, a(n,1)} is a group of real numbers instead of a single real number. In order to apply a fuzzy controller, we must the convert real number in the interval [0, 1]. Here, the proposed method uses a product operator as follows.

$$f(s_1, s_2, \cdots, s_n) = a(1,s_1) * a(2,s_2) \cdots * a(n,s_n). \tag{10}$$

A new vector u is defined as

$$\mathbf{u} = [f(1,1,\cdots,1), f(2,1,\cdots,1), \cdots, f(k_1,k_2,\cdots,k_n)] \tag{11}$$

where k_1, k_2, ⋯, k_n is the numbers of the membership functions for the inputs X_1, X_2, ⋯, X_n, and the i-*th* componet of **u** implies the firing strength of the i-*th* fuzzy rule.

We can computerm bold u and the relation matrix by using (12) and the defuzzified output b_1, b_2, ⋯, b_m are obtained by (13) using the moment method [4][6].

$$\mathbf{y} = \mathbf{u} \cdot (b_{i,j}) \tag{12}$$

where $\mathbf{y} = \{y_1, y_2, \cdots, y_{h_s}\}$, i = 1, 2, ⋯, k and j = 1, 2, ⋯, h_s, ($b_{i,j}$) denotes the relation matrix.

$$b_j = \sum_{j=1}^{h_1} (f_{ij} \times y_{\sum_{k-1}^{i} h_{k-1}+j}) / \sum_{j=1}^{h_i} y_{\sum_{k=1}^{i} h_{k-1}+j} \tag{13}$$

where h_i denotes the number of linguistic descriptions of the output variable Y_i. Note that h_0 is zero. And, f_{ij} is the central value of j-*th* membership function of the linguistic description for the output Y_i, where i=1, 2, …, m and j=1, 2, …, h_i.

IV. Trajectory optimization and control for robot manipulator

We now apply our methods to a real robot manipulator. For the experiments, we have chosen a SCARA-type robot whose first 2 joints are only used for simplicity. Fig. 2 depicts the configuration of the robot system. The robot system is composed of the SCARA-type robot, a DSP system and an IBM-PC.

1. Trajectory optimization for robot manipulator

The trajectory is constructed by a sequence of cubic polynomials. Each time interval di between the given knot points should be optimized so that the total traveling time may be minimized under the constraints. To find the optimal time intervals di's, we use the modified ES.

The knot points and the physical constraints for the robot are given in Tables 1 and 2.

Figs. 3 and 4 show that we can obtain the near minimum-time trajectories after 130 generations. The traveling time of the trajectories decreases as the number of generation increases. The minimized traveling time is 0.92 seconds, while the initial traveling time is 10 seconds. Figs. 5, 6 and 7 indicate that the optimized trajectory does not violate the given constraints. The performance comparison of the modified ES to the flexible polyhedron search method is shown in Fig. 8. In each trial, the search parameter vector is initialized. From the figure, our ES is proved to be better than the flexible polyhedron search method during 50 trials. We can expect such results, because ES is a global search method.

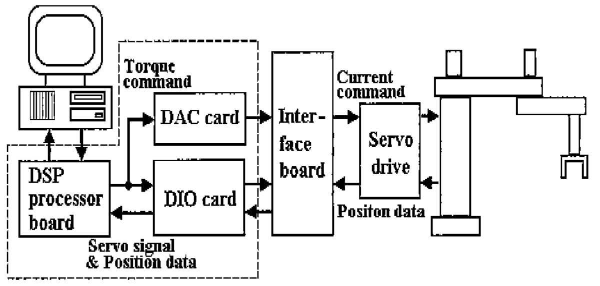


Fig. 2. Configuration of the robot system.

Table 1. Given knot points for the robot.

Knot point	Joint	
	Joint 1	Joint 2
Point1	0 rad	$\pi/2$ rad
Point3	$\pi/12$ rad	$5\pi/12$ rad
Point4	$5\pi/12$ rad	$\pi/12$ rad
Point6	$\pi/2$ rad	0 rad

Table 2. Constraints of the robot

Constraints	Joint	
	Joint 1	Joint 2
Velocity	$(5\pi/6)$ rad/sec	$(5\pi/6)$ rad/sec
Acceleration	$(20\pi/3)$ rad/sec^2	$(70\pi/9)$ rad/sec^2
Jerk	$(350\pi/9)$ rad/sec^3	50π rad/sec^3

2. Tracking control for robot manipulator using MIMO fuzzy reasoning method

Once the minimum-time trajectory is determined, we have to drive the manipulator precisely along the trajectory. The manipulator controller based on the MIMO fuzzy reasoning method is optimized to exactly track the trajectory by using the modified ES. In this case, the fitness function in (4) is defined as an inverse of the sum of tracking errors.

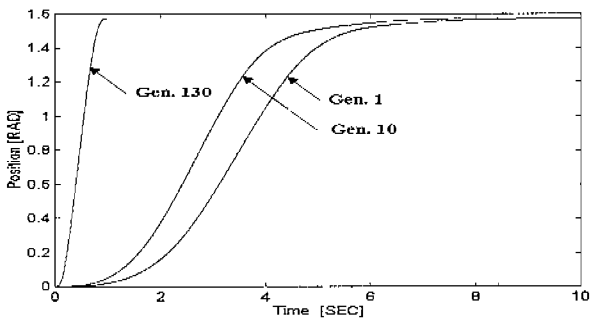


Fig. 3. Optimized position trajectory of link 1.

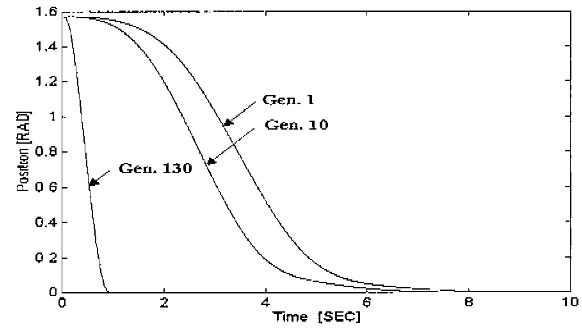


Fig. 4. Optimized position trajectory of link 2.

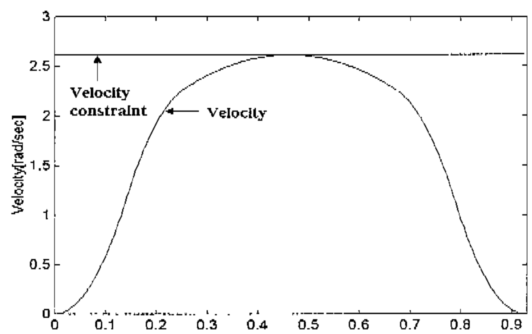


Fig. 5. Optimized velocity trajectory of link 1.

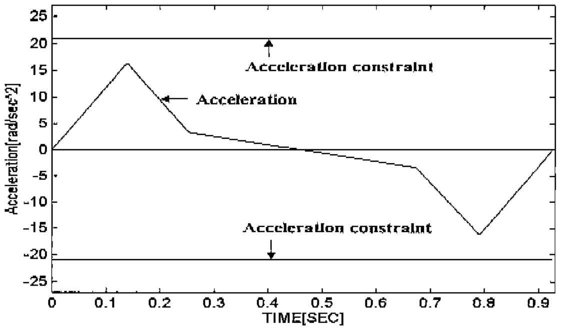


Fig. 6. Optimized acceleration trajectory of link 1.

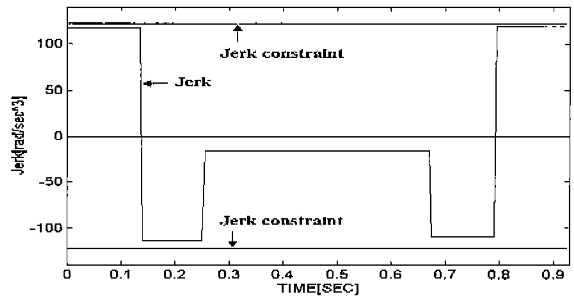


Fig. 7. Optimized jerk trajectory of link 1.

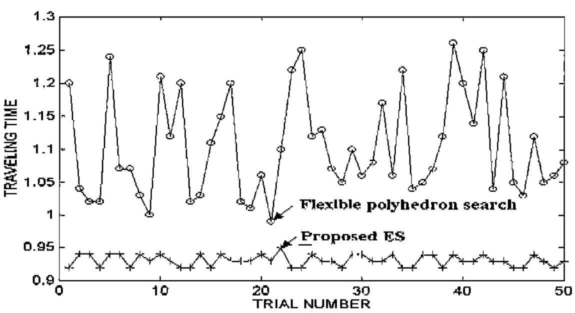


Fig. 8. Performance comparison of the modified ES and the flexible polyhedron search.

The input variables are the error and the error derivative, whereas the output variable is torque. The membership functions of both input and output variables are composed of N(negative) and P(positive). And Fig. 9 shows the membership function of input and output variables. Thus, the number of fuzzy rule sets is 16. The fuzzy relation matrix becomes a 16×4 matrix as shown below.

$$\begin{array}{c} e_1\ \dot{e}_1\ e_2\ \dot{e}_2 \\ (N,N,N,N) \\ (N,N,N,P) \\ \vdots \\ (P,P,P,P) \end{array} \quad \begin{array}{c} \quad\tau_1 \qquad\qquad \tau_2 \\ N \quad\ P \qquad N \quad\ P \\ \begin{pmatrix} b_{1,1} & b_{1,2} & b_{1,3} & b_{1,4} \\ b_{2,1} & b_{2,2} & b_{2,3} & b_{2,4} \\ \vdots & \vdots & \vdots & \vdots \\ b_{16,} & b_{16,2} & b_{16,3} & b_{16,4} \end{pmatrix} \end{array} \tag{14}$$

The individuals of the modified ES are made of the elements of the fuzzy relation matrix. The numbers of population and generation for the modified ES are 4 and 15, respectively.

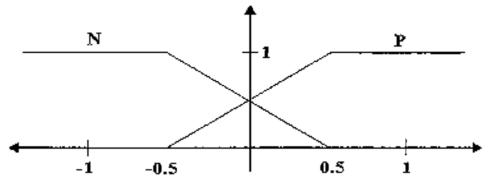


Fig. 9. Membership function of input and output variables.

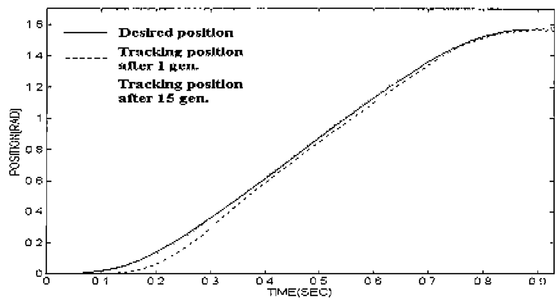


Fig. 10. Tracking position of link 1 after 1 and 15 generations.

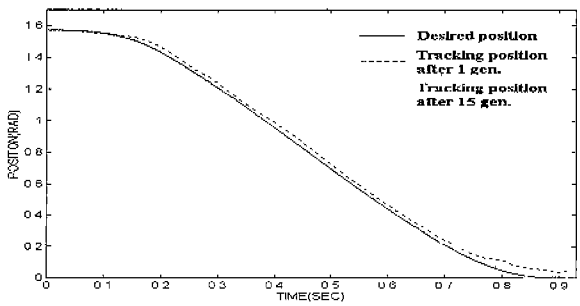


Fig. 11. Tracking position of link 2 after 1 and 15 generations.

The elements of the fuzzy relation matrix are optimized to reduce tracking error by using the modified ES. Although the large tracking error exists in the first generation, the manipulator can track the trajectory well after the 15-*th* generation. Figs. 10 and 11 show the tracking performance of the MIMO fuzzy controller after 1 and 15 generations.

The control performance becomes improved and the relation matrix is well tuned as the number of generation increases.

V. Conclusions

In this paper, we proposed a method for the optimal trajectory control of a robot manipulator using the evolution strategy. The method is composed of two steps. In the first step, we determine the optimal trajectory based on the cubic polynomials under some physical constraints. In the second step, the relation matrix of the MIMO fuzzy controller is optimized to precisely track the trajectory determined in the first step. Experimental results for a SCARA-type robot manipulator show that the proposed method is quite useful.

References

[1] Chun-Shin Lin, Po-Rong Chang and J. Y. S. Luh, "Formulation and optimization of cubic polynomial joint trajectories for industrial robots," *IEEE Trans. Automat. Contr.*, vol. 21, pp. 1066-1074, 1983.

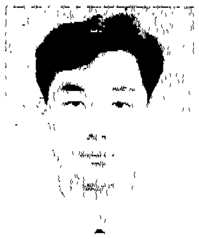
[2] David M. Himmelblau, Applied Nonlinear Programming, McGraw-Hill, pp. 148-157, 1972.

[3] Jin-Hyun Park and Young-Kiu Choi "An on-line PID control scheme for unknown nonlinear dynamic systems using evolution strategy," *Proceedings of IEEE International Conference on Evolutionary Computation*, pp. 759-763, 1996.

[4] Zhiqiang Cao, Abraham Kandel and Lihong LI, "A new model of fuzzy reasoning," *Fuzzy Sets and Systems*, vol. 36, pp. 311-325, 1990.

[5] K. S. Fu, R. C. Gonzalez and C. S. G. Lee, Robotics : Control, Sensing, Vision and Intelligence, McGraw-Hill, pp. 191-196, 1987.

[6] Daihee Park, Abraham Kandel and Gideon Langholz, "Genetic-Based New Fuzzy Reasoning Models with Applications to Fuzzy Control," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 24, no. 1, January, 1994.



Jin-Hyun Park

He received the B.S., M.S. and Ph.D. degrees in electrical engineering from Pusan National University, Korea, in 1992, 1994 and 1997, respectively. He was a full-time instructor at the department of electronics, Pusan College of Information Technology, from 1997 to 1999. He is currently a full-time instructor at the department of industrial automation engineering, Chinju National University. His research interests are intelligent control, evolutionary algorithms and robotics.

Young-Kiu Choi

He received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1980, and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology(KAIST), Taejon, in 1982 and 1987, respectively. He was a visiting scholar at CALTECH, U.S.A., from 1990 to 1991. Since 1998, he has been a visiting professor at University of Southwestern Louisiana, U.S.A. He is currently a professor at the department of electrical engineering, Pusan National University. His research interests are intelligent control, evolutionary algori thms, variable structure control, robotics and power electronics.

Hyun-Sik Kim

He received the B.S. and M.S. degrees in electrical engineering from Pusan National University, Korea, in 1994 and 1996, respectively. Since 1996, he has been a Ph.D. student in the department of electrical engineering, Pusan National University. He is currently a researcher in the Agency for Defense Development. His research interests are intelligent control, evolutionary algorithms, robotics and underwater vehicle autopilot.