'90 KACC 1990. 10. 26~27

A METHOD OF MINIMUM-TIME TRAJECTORY PLANNING ENSURING COLLISION-FREE MOTION FOR TWO ROBOT ARMS

Jihong Lee° and Zeungnam Bien

Dept. of Electrical Engineering, KAIST

P. O. Box 150, Chongyangni, Seoul, KOREA

ABSTRACT

A minimum-time trajectory planning for two robot arms with designated paths and coordination is proposed. The problem considered in this paper is a subproblem of hierarchically decomposed trajectory planning approach for multiple robots : i) path planning, ii) coordination planning, iii) velocity planning. In coordination planning stage, coordination space, a specific form of configuration space, is constructed to determine collision region and collision-free region, and a collision-free coordination curve (CFCC) passing collision-free region is selected. In velocity planning stage, normal dynamic equations of the robots, described by joint angles, velocities and accelerations, are converted into simpler forms which are described by traveling distance along collision-free coordination curve. By utilizing maximum allowable torques and joint velocity limits, admissible range of velocity and acceleration along CFCC is derived, and a minimum-time velocity planning is calculated in phase plane. Also the planning algorithm itself is converted to simple numerical iterative calculation form based on the concept of neural optimization network, which gives a feasible approximate solution to this planning problem. To show the usefulness of proposed method, an example of trajectory planning for 2 SCARA type robots in common workspace is illustrated.

1. INTRODUCTION

When robots are applied to do some tasks, it is found that planning is as much essential as control. In planning stage, we should define the overall tasks of each robot quantitatively, and then plan the trajectory of each robot by taking the capacities of the robots into consideration.

In many cases, tasks can be done in shorter time or more efficiently by multiple robots than by single robot. But when the task needs two robots in common workspace, for examples, it is very complex to derive collision-free motion for two robots, because one robot effectively becomes unpredictable moving obstacle to another. Due to such complexities relatively few algorithms have been proposed for the case of multiple robots.

Previous works related to multiple robot system can be classified into two groups ; i) local modification of trajectories of specific robots after independent planning[1,2,4], ii) simultaneous planning. To ensure global optimality, simultaneous planning is encouraged, but due to the complexity of the problem, a complete solution for general case is not reported yet. A simultaneous on-line planning method in local optimum sense was proposed in the work of Lee and Bien[3], and a hierarchically decomposed solution approach was taken by Shin and Bien[5]. Shin and Bien proposed 3 level decomposition method ; a) path planning, b) coordination planning, c) velocity planning. Each decomposed subproblem is not only an element of a feasible solution approach for general trajectory planning problem but also an independent problem according to tasks. Moreover they deviced a tool called "Coordination Space" to check collision between robots, and converted planning problem into simple geometrical problem. We deal with the third subproblem as independent problem, i.e., the case in which the paths and the coordination of two robots are given.

More specifically, it is proposed in the work of Tank and Hopfield[6] that the neural optimization network can solve such complicated linear programming problem instead of traditional analytical methods. Tsutgumi and Matsumoto[7] applied this neural optimization network concept in the planning of single robot with obstacles by modifying the concept into the form applicable to nonlinear programming problem.

In this paper, we consider a feasible approach of finding minimum-time velocity planning with given paths and coordination for two robots under the constraints in joint torques and joint velocities of each robot. After deriving a planning method analytically, we also propose an approximate solution process using neural optimization network.

2. NEURAL OPTIMIZATION NETWORK

There are many algorithmic methods for solving constrained minimization problems, and, of course, the neural optimization network method is one of them. The model of a neural network used in this work is a highly interconnected one of analog processors, and is called "Neural Optimization Network"[6]. This network can solve a linear programming problem and its basic operation is described below.

In the work of Tank and Hopfield, given M scalars B_j , $j=1,2,\cdots,M$, and the N dimensional vectors **A** and $\mathbf{D}_j$, $j=1,2,\cdots,M$, the following problem is considered:

[P] Determine an N dimensional vector **V** which minimizes the nonnegative objective function

$$\pi = \mathbf{A} \cdot \mathbf{V} \tag{1}$$

under the constraints

$$\mathbf{D}_j \cdot \mathbf{V} \geq B_j \qquad j=1,2,\ldots,M \tag{2}$$

where "·" denotes the inner product of two vectors.

By introducing an energy function in which objective function and constraint function are combined, original constrained minimization problem on R^n which is described by equation (1) and (2) is converted to an augmented but unconstrained minimization problem. Nextly a differential equation of variable V which satisfy equation (3) is derived.

$$\frac{dE}{dt} \leq 0 \quad t \in (0,\infty) \tag{3}$$

Then a network which operates under that differential equation makes that energy smaller and smaller, and finally gives V which makes that energy to smallest value, i.e., a solution of minimization problem.

3. COORDINATION SPACE AND COLLISION REGION

To efficiently handle collision between two robots, a concept called "Coordination Space (CS)" was proposed by Shin and Bien[5] , and we summarize the essence in the following.

When the paths of the robots are given, kinematics of each robots can be described by traveling distances along the paths for the case in which we can exclude redundancy of the robots. Hence they defined a 2-dimensional space in which two axis are the traveling distances along each path. Note that a point of CS corresponds to some configurations of two robots. Then a region in CS composed of all the points of CS which correspond to the configurations of two robots in collision state, is called "Collision Region (CR)". Details of determining CR may be found in the work of Shin and Bien. And a continuous curve connecting point (0,0) and ending point in CS without passing inside of collision region is called a "Collision-Free Coordination Curve (CFCC)". Then any velocity planning for the robots along a CFCC ensures collision-free motion. The CS and CR for the paths of Figure 1 is calculated and given in Figure 2.

4. MINIMUM-TIME VELOCITY PLANNING

A minimum-time velocity planning method with given coordination planning is described in this section. We summarize problem as follows :

[P] When a task is given such that two robot, whose kinematics and dynamics are given by (4)-(5), move from their corresponding initial points to final points along the designated path of (6) under a coordination of (7), find a pair of minimum-time velocity profile of each robot under given constraints of torque and velocity of actuators of two robots which are described by (8) and (9).

$$\mathbf{X}^r = \mathbf{K}^r(\mathbf{q}^r) \tag{4}$$

$$\mathbf{D}^r(\mathbf{q}^r)\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}^r, \dot{\mathbf{q}}^r) + \mathbf{g}(\mathbf{q}^r) = \mathbf{u}^r \tag{5}$$

$$\mathbf{x}^r = \boldsymbol{\psi}^r(s^r) \tag{6}$$

$$s^2 = \Gamma(s^1) \tag{7}$$

$$\mathbf{u}^r_{\min}(\mathbf{q}^r, \dot{\mathbf{q}}^r) \leq \mathbf{u}^r \leq \mathbf{u}^r_{\max}(\mathbf{q}^r, \dot{\mathbf{q}}^r) \tag{8}$$

$$\dot{\mathbf{q}}^r_{\min}(\mathbf{q}^r) \leq \dot{\mathbf{q}}^r \leq \dot{\mathbf{q}}^r_{\max}(\mathbf{q}^r) \quad r=1,2, \tag{9}$$

The coordination which is described by equation (7) determines a relation between two robots, hence with a coordination two robots operate under a synchronized manner. The coordination may be given for some task, for example, in which two robots carry a long bar grasping each end respectly, or may artificially be determined under some criterion. For example, the following performance was taken in [8], in which safety and degree of accomplishment are summed by weighting:

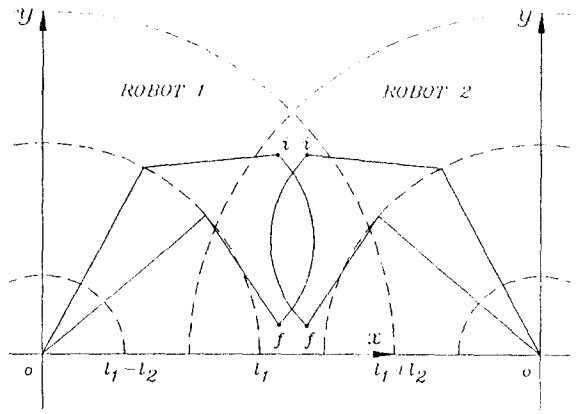

Fig.1. A two robot system in common workspace

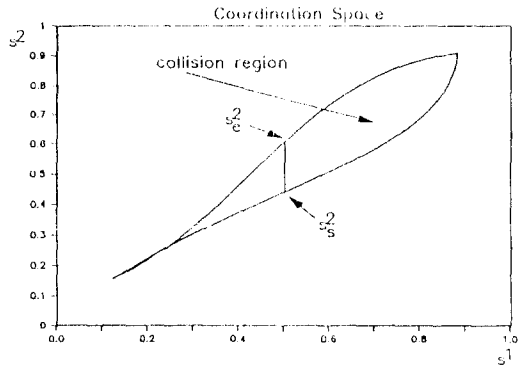

Fig.2. An example of coordination space and collision region

$$J_1 = \frac{\int_{cc} G(s^1, s^2)\, d\tau}{\int_{cc} d\tau}$$

$$G(s^1, s^2) = R/|d(s^1, s^2)| + Q|s^1 - s^2| \tag{10}$$

To solve the previously mentioned problem, firstly we define new variable "s", which is traveling distance along CC from (0,0) in CS. Then joint variables of the robots are described as a function of this variable such as

$$q^r_i = h^r_i(s) \quad , \quad r=1,2 \quad i=1,2,\ldots,N^r \tag{11}$$

where N^r is the total number of joint of robot r. Differentiating with respect to time we get

$$\dot{q}^r_i = \frac{dh^r_i}{ds}\dot{s} \tag{12}$$

$$\ddot{q}^r_i = \frac{d^2h^r_i}{ds^2}(\dot{s})^2 + \frac{dh^r_i}{ds}\ddot{s} \tag{13}$$

Also the dynamics for each joint of the robots is converted to

$$
\begin{aligned}
u_i^r &= \sum_{j=1}^{N^r} J_{ij}^r \ddot{q}_j^r + \sum_{j=1}^{N^r}\sum_{k=1}^{N^r} C_{ijk}^r \dot{q}_j^r \dot{q}_k^r + g_i^r \\
&= \sum_{j=1}^{N^r} J_{ij}^r \left[\frac{d^2 h_i^r (\dot{s})^2}{ds^2} + \frac{dh_i^r}{ds}\ddot{s} \right] \\
&\quad + \sum_{j=1}^{N^r}\sum_{k=1}^{N^r} C_{ijk}^r \frac{dh_j^r}{ds}\frac{dh_k^r}{ds}(\dot{s})^2 + g_i^r \\
&= \sum_{j=1}^{N^r} J_{ij}^r \frac{dh_i^r}{ds}\ddot{s} \\
&\quad + \left[\sum_{j=1}^{N^r} J_{ij}^r \frac{d^2 h_i^r}{ds^2} + \sum_{j=1}^{N^r}\sum_{k=1}^{N^r} C_{ijk} \frac{dh_j^r}{ds}\frac{dh_k^r}{ds} \right](\dot{s})^2 + g_i^r \\
&= M_i^r \ddot{s} + Q_i^r (\dot{s})^2 + g_i^r \qquad (14)
\end{aligned}
$$

Applying allowable torque bounds, the following inequalities are derived and arranged as

$$u_{i,min}^r \le M_i^r \ddot{s} + Q_i^r(\dot{s})^2 + g_i^r \le u_{i,max}^r \qquad (15)$$

$$
\begin{aligned}
&u_{i,min}^r - Q_i^r(\dot{s})^2 + g_i^r \le M_i^r \ddot{s} \\
&\le u_{i,max}^r - Q_i^r(\dot{s})^2 + g_i^r \qquad (16)
\end{aligned}
$$

In general case $u_{i,min}^r = -u_{i,max}^r$ is feasible assumption, hence (16) becomes

$$
\frac{-sign(M_i^r)u_{i,max}^r - Q_i^r(\dot{s})^2 + g_i^r}{M_i^r} \le \ddot{s} \le \frac{sign(M_i^r)u_{i,max}^r - Q_i^r(\dot{s})^2 + g_i^r}{M_i^r} \qquad (17)
$$

If we define new variable $v = \dot{s}$, then (17) is described in simpler form as

$$LB_i^r \le \dot{v} \le UB_i^r \qquad (18)$$

And (17) should be satisfied for all joints of the robots, therefore we get

$$\underset{i,r}{Max}\ LB_i^r \le \dot{v} \le \underset{i,r}{Min}\ UB_i^r \qquad (19)$$

and describe it in simple form

$$LB(s) \le \dot{v} \le UB(s) \qquad (20)$$

To be physically meaningful, the solution of $\dot{v}$ in (20) should exist. Hence from (19)

$$\underset{i,r}{Min}\ UB_i^r \ge \underset{i,r}{Max}\ LB_i^r \qquad (21)$$

and these inequalities mean for each joint

$$UB_i^p - LB_j^q \ge 0 \qquad (22)$$

$$p=1,2,\ q=1,2\ ,\ i=1,2,\ldots,N^p\ ,\ j=1,2,\ldots,N^q$$

Then by substituting original form of (17) for (22), we get

$$\left[\frac{Q_i^p}{M_i^p} - \frac{Q_j^q}{M_j^q} \right] v^2 + \left[\frac{u_{i,max}^p - g_i^p}{|M_i^p|} - \frac{u_{j,max}^q - g_j^q}{|M_j^q|} \right] \ge 0$$

$$p=1,2,\ q=1,2\ ,\ i=1,2,\ldots,N^p\ ,\ j=1,2,\ldots,N^q \qquad (23)$$

The solution of (23) is calculated in such form as

$$0 \le v \le \phi_1(s) \qquad (24)$$

because we exclude reverse motion along CFCC. Also from the maximum allowable joint velocity of (9), another set of inequalities are derived by following calculation. With arbitrary value of v_0, corresponding joint velocities are calculated.

$$\dot{q}_{i,0}^r = \frac{dh_i^r}{ds} v_0 \qquad (25)$$

Then scaling factor is calculated to adjust the calculated joint velocities to maximum allowable limits.

$$\gamma = \max_{i,r} \left[\frac{|\dot{q}_{i,0}^r|}{|\dot{q}_{i,max}^r|} \right] \qquad (26)$$

And scale the previously used velocity v_0 by scaling factor.

$$v_{\max} = \frac{v_0}{\gamma} \qquad (27)$$

The result may be arranged in inequality form :

$$0 \le v \le \phi_2(s) \qquad (28)$$

From the inequalities of (24) and (28), the allowable velocities along coordination curve are arranged in form of

$$0 \le v \le Min(\phi_1(s), \phi_2(s)) = \phi(s) \qquad (29)$$

An example of $\phi(s)$ is given in Figure 3. And the total traveling time along CC is

$$T_f = \int dt = \int_0^S \frac{dt}{ds} ds = \int_0^S \frac{1}{\dot{s}} ds = \int_0^S \frac{1}{v} ds \qquad (30)$$

where S is total traveling distance along CC from (0,0) to ending point in CS. From (30) we can know by intuition that minimum-time velocity planning should sustain its maximum value at every instant without violating accelera-

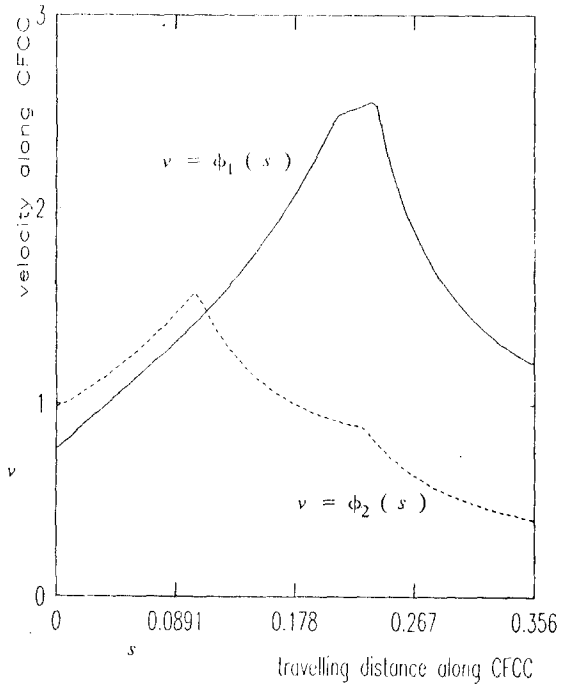

Fig.3. An example of phase plane

tion limit. At this point, we describe a minimum-time velocity planing algorithm based on the method for single robot case in [9,10].

s1) From $s=0$, $v=v_f$, construct a trajectory with maximum acceleration UB in forward direction until this curve meets $v=\phi$ or goes past $s=S$.

s2) From $s=S$, $v=v_f$, construct a trajectory with maximum deceleration LB in backward direction until this curve meets $v=\phi$ or goes past $s=0$.

s3) If two curves intersect, stop. Call this point s_1, then s_1 is switching point.(Figure 4)

s4) If two curves do not intersect, forward curve must meet the curve $v=\phi$ at some value of s. Call this point s_2.(Figure 5)

s5) At $s=s_2$, calculate the value of $d\phi/dt$, then three cases are possible.

s5a) If $d\phi/dt$ is smaller than LB, follow the curve $v=\phi$ from $s=s_2$, until reaching a point $s=s_3$ at which $d\phi/dt$ equals to LB. Construct a curve that starts at $s=s_3$, $v=\phi(s_3)$ with LB in backward direction until it meets previously constructed trajectory. Go to s6).

s5b) If $d\phi/dt$ is within the range between LB and UB, follow the curve $v=\phi$ from s_2, until reaching a point $s=s_4$ at which the value of $d\phi/dt$ equals to LB or UB or this curve meets final backward trajectory. If it meets backward trajectory, stop. If $d\phi/dt = UB$ go to s6). If $d\phi/dt = LB$ go to s5a).

s5c) If the value of $d\phi/dt$ at s_2 is greater than UB, go to s6).

s6) Construct a trajectory with maximum acceleration in forward direction which starts from current point until it meets a final deceleration curve or the curve $v=\phi$. If it meets a final deceleration curve, stop. If it meets the curve $v=\phi$, go to s5).

All the intersection points of the trajectories are switching point and this algorithm gives a sequence of curve segments which are under maximum acceleration or maximum deceleration or identical to maximum allowable velocity curve. The optimality in the sense of minimum-time is obvious by noting the following : any trajectory of less time has an interval of higher velocity than the trajectory calculated by above algorithm, but that is impossible without violating the constraint of acceleration bound, because it needs more acceleration or more deceleration than permitted. Detail proof is omitted here.

The proposed algorithm gives exact solution for given problem, but it is somewhat complex and time consuming. Hence as an alternative we propose an approximate iterative calculation version by discretizing the variable s by the interval of Δs. Then admissible velocity range and acceleration is described in discretized form ;

$$0 \le \dot{s}_k = v_k \le \phi(k\Delta s) \tag{31}$$

$$LB(k\Delta s) \le \ddot{s}_k = \dot{v}_k \le UB(k\Delta s) \tag{32}$$

And also the accelerations are approximated as

$$\dot{v}_k = \frac{v_{k+1}^2 - v_k^2}{2\Delta S} \tag{33}$$

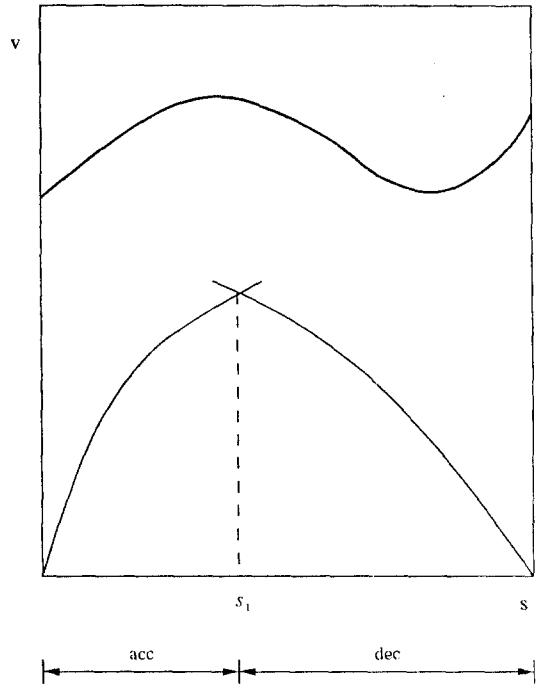

Fig.4. Case when accelerating and decelerating curves intersect

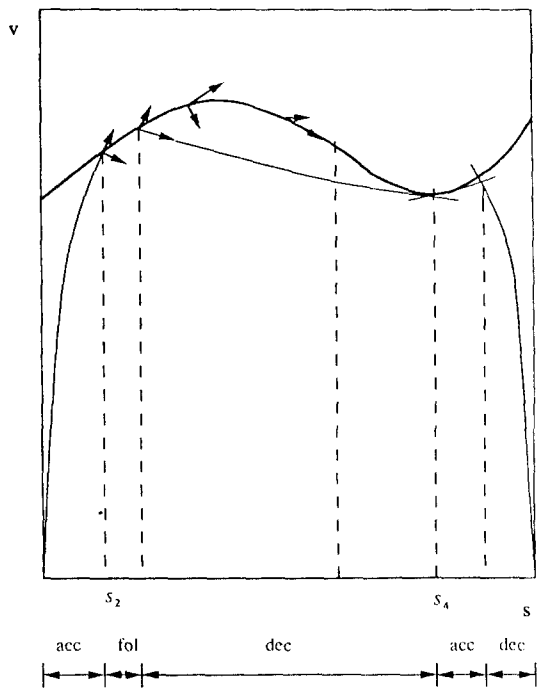

Fig.5. Case when accelerating and decelerating curves do not intersect

At this point we define each energy as

$$E_1 = \frac{1}{2} \sum_{k=1}^{N-1} (\phi(k\Delta s) - v_k)^2 \tag{34}$$

$$E_2 = \sum_{k=0}^{N-1} F\left[\frac{v_{k+1}^2 - v_k^2}{2\Delta s} - UB(k\Delta s)\right] + \sum_{k=0}^{N-1} F\left[LB(k\Delta s) - \frac{v_{k+1}^2 - v_k^2}{2\Delta s}\right] \tag{35}$$

$$F(z) = \begin{cases} z^2/2 & z > 0 \\ 0 & z \le 0 \end{cases} \tag{36}$$

and sum these with weighting

$$E = w_1 E_1 + w_2 E_2 \tag{37}$$

Introducing the concept of neural optimization network, we determine differential equation of each v_k as

$$\frac{dv_k}{dt} = -\frac{dE}{dv_k} = w_1 \sum_{k=1}^{N-1} (\phi(k\Delta s) - v_k) + w_2 \sum_{k=1}^{N-1} f\left[\frac{v_{k+1}^2 - v_k^2}{2\Delta s} - UB(k\Delta s)\right]\frac{v_k}{\Delta s} - w_2 \sum_{k=1}^{N-1} f\left[LB(k\Delta s) - \frac{v_{k+1}^2 - v_k^2}{2\Delta s}\right]\frac{v_k}{\Delta s}$$

$$f(z) = \begin{cases} z & z > 0 \\ 0 & z \le 0 \end{cases} \tag{38}$$

Then time derivative of total energy becomes

$$\frac{dE}{dt} = \sum_{k=1}^{N-1} \frac{dE}{dv_k}\frac{dv_k}{dt} = -\sum_{k=1}^{N-1} \left[\frac{dv_k}{dt}\right]^2 \le 0 \tag{39}$$

till it reaches local minimum. We apply this idea to the following robot system with the paths of Figure 1.

$$\begin{aligned} &-25\ Nm \le u_1^r \le 25\ Nm \\ &-7\ Nm \le u_2^r \le 7\ Nm \\ &-2\ rad/sec \le \dot{q}_1^r \le 2\ rad/sec \\ &-2.5\ rad/sec \le \dot{q}_2^r \le 2.5\ rad/sec \qquad r=1,2 \end{aligned} \tag{40}$$

By consulting Figure 2 , we selected a CFCC as

$$s_2 = (s_1)^2 \tag{41}$$

and initial v_k as its largest value as

$$v(k\Delta s) = \phi(k\Delta s) \quad , \quad k=1,2,\ldots, N \tag{42}$$

The result is presented in Figure 6 and the joint velocities corresponding to finally settled solution are in Figure 7. It is noteful that the result is same to that of [5], in which dynamic programming technique is applied under same conditions.

5. CONCLUDING REMARKS

A minimum-time trajectory planning method for multiple robots is presented. Specifically the case in which paths and coordination are given is considered. After deriving a planning method analytically, an iterative numerical solution approach based on the concept of neural optimization network is applied.

The method presented in the paper needs further refinements, especially to handle the local minimum problem, and to extend this method to more general cases.

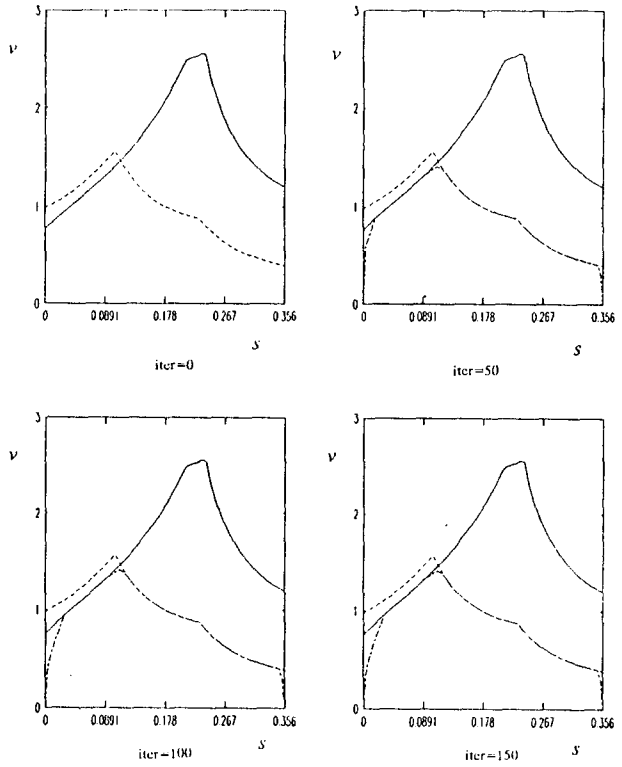

Fig.6. Modification of velocity profile

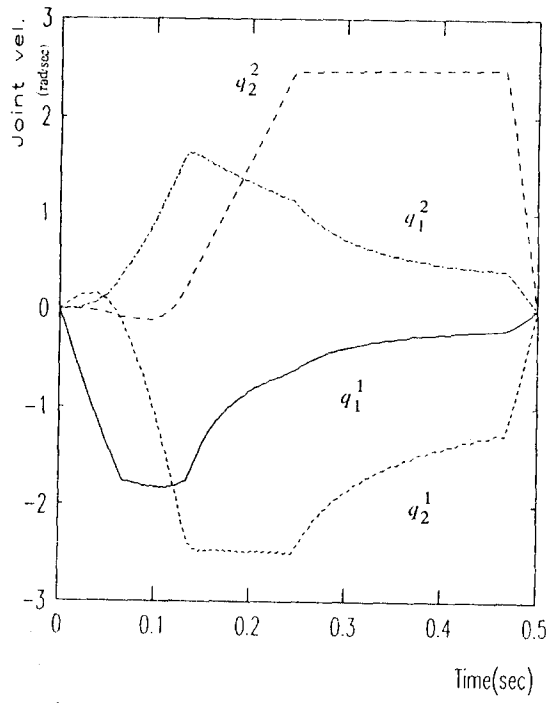

Fig.7. Minimum-time joint velocities

REFERENCES

1. B. H. Lee and C. S. G. Lee, "Collision-Free Motion Planning of Two Robots", *IEEE Trans. on SMC*, Vol. SMC-17, No. 1, pp. 21-32, Jan/Feb. 1987.
2. E. Freund and H. Hoyer, "Real-Time Pathfinding in Multirobot Systems Including Obstacle Avoidance", *Int'l J. of Robotics Research* Vol. 7, No. 1, pp. 42-70, Feb. 1988
3. J. Lee and Z. Bien, "Collision-Free Trajectory Control for Multiple Robots based on Neural Optimization Network", *Robotica* Accepted for publication 1990
4. C. Chang, M. J. Chung and Z. Bien, "Collision-free motion planning for two articulated robot arms using minimum distance functions", *Robotica* Accepted for publication 1990
5. Y. Shin and Z. Bien, "Collision-free trajectory planning for two robot arms", *Robotica* Vol. 7, pp.205-212, 1989
6. D. W. Tank and J. J. Hopfield, "Simple "Neural" Optimization Networks: An A/D Converter, Signal Decision Circuit, and a Linear Programming Circuit", *IEEE Trans. on Circuits and Systems*, Vol. CAS-33, No. 5, May 1986.
7. K. Tsutgumi and H. Matsumoto, "Neural computation and Learning Strategy for Manipulator Position Control", *International conference on Neural Networks* , Vol. IV, pp. 525, 1987.
8. Y. Shin, "Collision-Free Trajectory Planning for Two Robots", Dissertation for Ph. D, KAIST, Korea, 1989
9. J. E. Bobrow, S. Dubowski and J. S. Gibson, "Time-Optimal Control of Robotic Manipulators Along Specified Paths", *Int'l J. of Robotics Research*, Vol. 4, No. 3, pp.3-17, Fall 1985
10. K. G. Shin and N. D. McKay,"Minimum-Time Control of Robotic Manipulators With Geometric Path Constraints", *IEEE Trnas. Automatic Control*, Vol. AC-30, No. 6, pp.531-541, Jun 1985