論文 88－25－8－5

Motion Level Modifications for Collision Avoidance of Two Manipulators

(두 매니퓰레이터의 충돌 회피를 위한 동작단계 수정법)

李 範 熙*
(Bum Hee Lee)

要 約

본 논문은 두 매니퓰레이터의 충돌회피에 대한 최근 연구결과를 동작단계 수정법들을 중심으로 다룬다. 가정되는 충돌상황은 한 로보트가 다른 로보트와의 충돌회피 목적상 미리 정해진 최종도달시간 K_f와 미리 정해진 경로를 변경할 수 있는 경우이다. 조직적인 접근을 위해 세개의 독립적인 세분화를 통해 충돌회피 문제가 다루어진다.

Abstract

This paper presents the recent findings for collision avoidance of two manipulators in terms of motion level modifications. The collision situation we assume here is that the prespecified final time K_f and the prespecified path of one robot can be modified for the purpose of collision avoidance with the other robot. The collision avoidance problem is resolved into three independent categories for a systematic approach.

I. Introduction

Motion planning is composed of path planning and trajectory planning of the robot system. Collision-free motion planning is achieved through collision-free path planning schemes. Most of the path planning schemes concern the problem of avoiding fixed and stationary obstacles in a workspace. Due to the fixed and stationary obstacles, the path planning problem is usually converted to a geometric analysis problem for obtaining a collision-free path.

It is interesting to note that there are several experimental works on obstacle avoidance. Gouzenes [4] discussed collision avoidance of manipulators in a flexible assembly cell, using graph-search techniques and Petri nets. Myers et al. [7] used a fast static collision check for detecting potential collisions with obstacles. They developed a heuristic method to determine a collision-free path in a reasonable amount of time and demonstrated an application on a VAX-11/780 computer and a microcomputer. It is comparable to the work done by Petrov [8]. Compared to conventional collision avoidance, the collision avoidance problem for two manipulators should be solved subject to the path and trajectory information of the two manipulators together.

*正會員, 서울大學校 制御計測工學科
(Dept. of Control and Instrumentation Eng., Seoul Nat'l Univ.)

接受日字：1988年 2月 6日

Recently, Fortune et al. [1] developed an O (n^3) algorithm for independent but synchronized motion of two Stanford arms. Tournassoud [9] presented also a local method for collision avoidance based on the existence of extreme separating hyperplanes between two manipulators. In particular, the collision avoidance problem associated with two manipulators in a common workspace can be found in Freund's work [2,3].

In this paper we represent each robot by a single sphere at the wrist. It is assumed that each robot follows a straight line trajectory faithfully which is provided by the algorithmic straight line trajectory planner in Lee [6]. Thus, each robot exactly follows a straight line path at all the servo time periods, which is very important in defining the robot speed in the direction of the straight line path. It is also assumed that no collisions occur at the initial and final locations of the two robots and the collisions are restricted to the wrist level of the robot.

II. Problem Classifications

The method in obtaining a collision-free path or trajectory may vary depending on various collision situations especially when two robots are moving on the straight line paths simultaneously. When two robots (we call them robot 1 and robot 2) are assumed to move on their planned straight line paths with potential collisions, various collision situations are identified as follows:

1. Case 1: The final arrival time k_f of robot 2 can be relaxed but its original path cannot be changed for the purpose of collision avoidance because the modification of path may induce potential collisions with other fixed obstacles.
2. Case 2: The final arrival time k_f of robot 2 can be relaxed and its original path can be changed for the purpose of collision avoidance.
3. Case 3: The fianl arrival time k_f of robot 2 cannot be relaxed but its original path can be changed for the purpose of collision avoidance.
4. Case 4: The final arrival time k_f of robot 2 cannot be relaxed and its original path cannot be changed for the purpose of collision avoidance.

In case 1, we can only change the speed or delay the motion of robot 2 along the original path to avoid the collision. From now on, the terminology collision will be used to denote the terminology potential wrist collision. Since the change of robot speed can only be accomplished by modifying the trajectory information, a procedure for a speed change needs to be developed to obtain a collision-free trajectory of robot 2. The delay of robot 2 motion can also be utilized for the purpose of collision avoidance. It then corresponds to the time coordinated, independent motion of the two robots [5].

In case 2, collision-free motion planning can be considered in the following categories with the preplanned minimum time trajectory:

1. category 1: When speed reduction and/or time delay of the robot 2 motion is applied without any path modification;
2. category 2: When only path modification is applied;
3. category 3: when path modfication with speed reduction and/or time delay of the robot 2 motion are applied simultaneously.

In category 1, a collision-free path can be found exactly in the same way as in case 1. If a solution from category 1 is not adequate because a fairly large time delay is required or speed reduction is not appropriate for the collision avoidance, then we can consider a solution in categories 2 and 3. In category 2, there are a variety of freedoms in choosing a collision-free path. The choice of the collision-free path depends on the environment of the workspace and various user designated performance indices. Sometimes, a solution in category 2 corresponds to a solution in category 3 due to various reasons, for example, a robot speed constraint, a path deviation constraint, etc. Clearly, the unnecessary path deviation in category 2 can be reduced by an appropriate delay along a collision-free path.

In case 3, due to the fixed final arrival time k_f, there is no guarantee for another collision-free path for robot 2 satisfying the final arrival time k_f. In case 4, the collision avoidance must be realized by changing the path and/or trajectory of the other moving robot.

As mentioned earlier, the results on the case 1 situation can be found in Lee [5], and the discussions on the case 3 and case 4 are not practically applicable in reality. Here, we investigate the case 2 collision situation through three independent categories.

III. Collision Avoidance for Case 2

A collision-free path of robot 2 is considered in the following categories: (1) when speed reduction or time delay is applied without any path modification, (2) when only path modification is applied without considering the trajectory information of two robots, and (3) when path modification and speed reduction or time delay are applied simultaneously.

1. Time Dealy

As indicated in Lee [5], time delay yields the shorter final arrival time than speed reduction for avoiding a potential collision. Thus, time delay is preferred in this paper. If we denote the final arrival time of robot 2 as k_f for the original trajectory, then the total traveling time after a time delay will be:

$$k_{TD} = k_f + \Delta k_f \tag{1}$$

where Δk_f is a required time delay for avoiding a potential collision. If Δk_f is very small compared to k_f, then time delay of the original trajectory provides us a good solution for the purpose of collision avoidance. When Δk_f is fairly large, we can consider a solution in categories (2) and (3). Also, since two robots are working simultaneously in a common workspace, and their movements are coordinated in a time sequence, there is likely to be a constraint on the time delay of the robot 2 motion.

2. Path Modification

First, we find a collision-free path which deviates from the robot 1 path for at least a distance of $r_1 + r_2$ through a geometric analysis, where r_1 and r_2 denote the radius of the sphere model for each robot. The path is then guaranteed for the collision avoidance with robot 1.

A collision situation is shown in Figure 1, where robot 1 moves from A_1 to B_1, while robot 2 moves from A_2 to B_2. We assume that a potential collision exists between the two moving robots. The following analysis indicates how to choose a point C_2 such that the path from A_2 to B_2 via C_2 deviates from the robot 1 path for at least a distance of $r_1 + r_2$, which guarantees the collision avoidance.

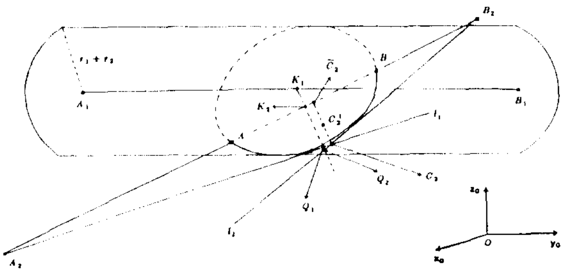


Fig.1. Collision avoidance by path modification.

We denote the nearest two points on the robot 1 and robot 2 path as K_1 (x_{K1}, y_{K1}, z_{K1}) and K_2 (x_{K2}, y_{K2}, z_{k2}), respectively, as shown in Figure 1. It is notable that points A_2, B_2, K_1, and K_2 are on the same plane which can be constructed by $\overline{A_2B_2}$ and $\overline{K_1K_2}$.

It is notable that the swept volume of a sphere in the direction of a straight line forms a cylinder capped with two semi-spheres at both ends. Thus, for considering the distance of $r_1 + r_2$ between the two robots, we view the collision situation between the cyclinder surface of radius $r_1 + r_2$ in the direction of A_1B_1 and the robot 2 path. There are two intersection points on the cylinder surface, we call them A and B, at which the cylinder surface is perforated by the robot 2 path.

To maintain the distance of $r_1 + r_2$ between the robots, we consider the interesecting curve between the cylinder surface and the plane which is constructed by $\overline{A_2B_2}$ and $\overline{K_1K_2}$. We call the intersecting curve on the cylinder surface a collision-free locus. The collision-free locus connecting A and B forms a part of an ellipse, which is centered at K_1.

In order to obtain straight line segments which approximate the collision-free locus, we construct lines ℓ_1 and ℓ_2 which lead separately from A_2 and B_2 and are tangent to the ellipse. Consider two points Q_1 and Q_2 in the direction of $\overline{K_1K_2}$. We denote their deviations from K_1 as d^1 and d^2. First, we obtain the location of Q_1 and Q_2 in terms of d^1 and d^2, respectively. Then, we find constraints on d^1 and d^2 to guarantee that the distances between two straight line paths A_1K_1 and A_2Q_1, and B_1K_1 and B_2Q_2 are equal to $r_1 + r_2$.

The location of Q_1 $(x_{Q_1}, y_{Q_1}, x_{Q_1})$ is obtained in terms of d^1 as:

$$\overrightarrow{OQ_1} = \overrightarrow{OK_1} + \frac{\overrightarrow{OK_2} - \overrightarrow{OK_1}}{\| \overrightarrow{OK_2} - \overrightarrow{OK_1} \|} . d^1 \tag{2}$$

Since the value of d^1 specifies the location of Q_1, we can represent the distance between two straight line paths A_1K_1 and A_2Q_1 in terms of the deviation d^1. If we denote this distance as DS_1, then:

$$DS_1 = f_1 (d^1) \tag{3}$$

where f_1 relates the distance between the paths of A_1K_1 and A_2Q_1 with the deviation d^1.

To maintain the distance between the paths A_1K_1 and A_2Q_1 for at least $r_1 + r_2$, we must have:

$$DS_1 = f_1 (d^1) \geqslant r_1 + r_2 \tag{4}$$

Here, we will consider the equality only to avoid an unnecessary large deviation from K_1. Then, we can obtain a constraint on the deviation d^1 as:

$$d^1 = f_1^{-1} (r_1 + r_2) \tag{5}$$

Similarly, for the paths of B_1K_1 and B_2Q_2, we have a constraint on the deviation d^2 as:

$$d^2 = f_2^{-1} (r_1 + r_2) \tag{6}$$

where f_2 relates the distance between the paths of B_1K_1 and B_2Q_2 with the deviation d^2. The location of Q_1 can be obtained from Eq. (2) by using d^1 of Eq. (5). Also, the location of Q_2 can be obtained from Eq. (2) by substituting d^2 of Eq. (6) for d^1. Using the locations of Q_1 and Q_2, the two straight lines ℓ_1 and ℓ_2 in the directions of $\overline{A_2Q_1}$ and B_2Q_2 can be identified easily. It is notable that ℓ_1 and ℓ_2 are on the same plane which is constructed by $\overline{A_2B_2}$ and $\overline{K_1K_2}$:

Now, we consider the intersection point of these two straight lines ℓ_1 and ℓ_2. Apparently, the path from A_2 to B_2 via the intersection point, which is denoted as C_2, is a collision-free path in category (2). To find the deviation of this path from the original robot 2 path, the nearest point on $\overline{A_2B_2}$ from C_2, which we call $\bar{C}_2$, is found from a vector projection and addition as:

$$\overrightarrow{O\bar{C}_2} = \overrightarrow{OB_2} + (\overrightarrow{OA_2} - \overrightarrow{OB_2}) . \frac{\overrightarrow{B_2C_2} . (\overrightarrow{OA_2} - \overrightarrow{OB_2})}{\| \overrightarrow{OA_2} - \overrightarrow{OB_2} \|^2} \tag{7}$$

Then the actual deviation of the collision-free path from the original robot 2 path is found as:

$$d^{act} = \| \overrightarrow{O\bar{C}_2} - \overrightarrow{OC_2} \| \tag{8}$$

Note that no time delay of the robot 2 motion is required for the path from A_2 to B_2 via C_2 for the purpose of collision avoidance, while Δk_f in Eq. (1) is required for the original robot 2 path. Of course the time of travel from A_2 to B_2 via C_2 will exceed K_f.

3. Time Delay with Path Modification

We now consider a collision-free path by both path modification and time delay. A path, which deviates from $\bar{C}_2$ for a distance of Δd, is considered. A point C_2^1 can be found for the deviation of Δd from $\bar{C}_2$ as:

$$\overrightarrow{OC_2^1} = \overrightarrow{O\bar{C}_2} + \frac{\overrightarrow{OC_2} - \overrightarrow{O\bar{C}_2}}{\| \overrightarrow{OC_2} - \overrightarrow{O\bar{C}_2} \|} \cdot \Delta d \tag{9}$$

The point C_2^1 specifies the lengths of $\overline{A_2C_2^1}$ and $\overline{C_2^1B_2}$ as $\bar{\ell}_1^1$ and $\bar{\ell}_2^1$, respectively. We now try to use the collision map to determine the required time delay for the purpose of collision avoidance.

As discussed in Lee [5], two moving robots have the potential for colliding with each other under the original trajectory information, if there is a range of collision lengths where the path of robot 2 is within the colliding range of a point on the path of robot 1. The union of these collision lengths at the collection of servo time instants determining the points on the path of robot 1 can be drawn as a connected region.

See Figure 2. More precisely, the collision length at time K corresponds to all points on the path for robot 2 that are within $r_1 + r_2$ of the point that lies on the path or robot 1 at time K. If the traveled length versus servo time curve for robot 2 (which is from robot 2 trajectory information) touches or crosses the region, it indicates that a potential collision of the wrists of robot 1 and robot 2 exists. Details regarding the method to obtain the collision maps can be found in Lee [5]. Considering Eq. (9), we have the corresponding collision map as shown in Figure 3, where the traveled length versus servo time curve OM^1 is assumed to cross the collision region (bounding box approximated region for convenience) for the path $A_2 \rightarrow C_2^1 \rightarrow B_2$. The required time delay for avoiding the collision region can be found from the trajectory information or calculated from the collision map roughly.

The traveled length versus servo time curve OM^2 is a curve, which is obtained from OM^1 with the required time delay for the path $A_2 \rightarrow C_2^1 \rightarrow B_2$.

As mentioned earlier, the two robots are moving simultaneously in a time-coordinated sequence. Other objects may need to move close to the initial location of robot 2. Thus, there is likely to be a constraint on the time delay at the initial location for avoiding the potential collision. If the allowable time delay at the initial location is denoted as Δk_{allow}, we want to obtain a collision-free path which does not violate this constraint. It is notable that the path from A_2 to B_2 via C_2 does not need any time delay for the purpose of collision avoidance. Thus, this path always meets the constraint on the time delay at the expense of the deviation d^{act}.

If Δk_f in Eq. (1) is smaller than or equal to Δk_{allow}, then the solution in category (1) will be enough for the purpose of collision avoidance. However, if Δk_f is larger than Δk_{allow}, then time delay on the original robot 2 path cannot be used for the purpose of collision avoidance. In this aspect, we want to find a collision-free path by both path modification and time delay of the robot 2 motion. Note, however, that although the time delay we determine will be less than Δk_{allow}, the total traveling time may be larger than K_{TD} in Eq. (1).

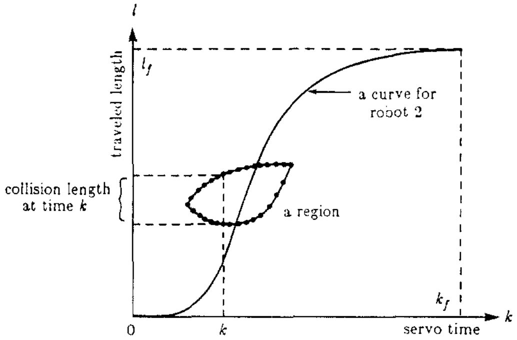


Fig.2. A primitive collision map.

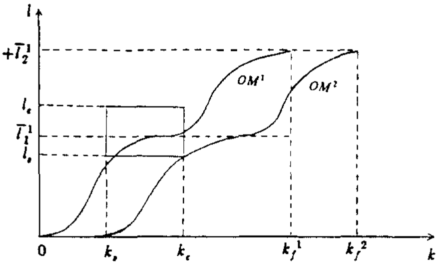


Fig.3. Collision map with path modification.

If the required time delay for a path between A_2B_2 and the path from A_2 to B_2 via C_2^1 exceeds Δk_{allow}, then we can increase the path deviation from the original robot 2 path. Otherwise, we can decrease the path deviation from the original robot 2 path. A bisection method between $\bar{C}_2$ and C_2 can be used to increase or to decrease the path deviation depending on whether the required time delay exceeds. Note that, since the path from A_2 to B_2 via C_2 does not need any time delay for the purpose of collision avoidance, the existence of a collision-free path from the bisection process is always guaranteed.

IV. Summary

The collision avoidance problem between two manipulators was considered for the case 2 collision situation, where the original path and trajectory of robot 2 can be modified for the purpose of collision avoidance. The problem was investigated through three different phases; time delay, path modification, and time delay with path modification. Together with the results in

Lee [5], this paper will constitute a keystone in collision avoidance planning of two manipulators.

References

[1] S. Fortune et al., "Coordinated Motion of Two Robot Arms," *Proc. IEEE Conf. on Robotics and Automation,* San Francisco, pp. 1216-1223, 1986.

[2] E. Freund, "Hierarchical Control of Guided Collision Avoidance for Robots in Assembly Automation," *Proc. of 4th Int'l Conf. on Assembly Automation,* pp. 91-103, September 1983. Japan

[3] E. Freund, "On the Design of Multi-Robot Systems," *Proc. of IEEE International Conference on Robotics and Automation Atlanta, Georgia,* March 13-15, pp.477-490.

[4] L. Gouzenes, "Collision Avoidance for Robots in an Experimental Flexible Assembly Cell," *Proc. of IEEE International Conference on Robotics and Automation Atlanta, Georgia,* March 13-15, pp. 474-476.

[5] B.H. Lee et al., "Collision-Free Motion Planning of Two Robots," *IEEE Trans. on System, Man and Cybernetics,* vol. 17, no.1, pp. 21-32, Feb. 1987.

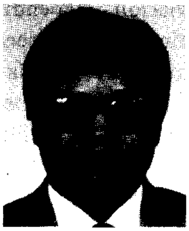
[6] B.H. Lee, "Algorithmic Approach to Straight Line Trajectory Planning for Mechanical Manipulators," Proceedings of 1986 American Control Conference, June 18-20, Seattle, Washington, pp. 121-126.

[7] J.K. Myers, et al., "A Supervisory Collision-Avoidance System for Robot Controllers," The Winter Annual Meeting of the American Society of Mechanical Engineers, November 14-19, Phoenix, Arizona, 1982, pp. 225-232.

[8] A.A. Petrov and I.M. Sirota, "Obstacle Avoidance by a Robot Manipulator under Limited Information about the Environment," Automatica, no. 4, Moscow, pp. 29-40, April 1983.

[9] P. Tournassoud, "A Strategy for Obstacle Avoidance and Application to Multi-robot Systems," *Proc. IEEE Conf. on Robotics and Automation,* San francisco, pp.1224-1229, 1986.

著者紹介



李 範 熙(正會員)

1955年 5月 29日生. 1978年 2月 서울대학교 전자공학과 졸업. 1980年 2月 서울대학교 대학원 전자공학과 공학석사학위 취득. 1985年 미국 Univ. of Michigan, 전기과 로보트공학 연구실 공학박사학위 취득. 1980年 3月～1981年 8月 중앙대학교 전자공학과 전임 강사. 1982年 5月～1985年 7月 미국 Univ. of Michigan, Center for Robotics and Integrated Manufacturing 연구원. 1985年 8月～1987年 1月 미국 Purdue Univ. School of Electrical Eng. 조교수 1987年 2月～현재 서울대학교 제어계측공학과 조교수. 주관심분야는 로보틱스 및 자동화 등임.