

두대의 회전 다관절형 로봇 매니퓰레이터를 위한 충돌회피 방법

장철^o, 정명진 이범희
한국과학기술원 전기 및 전자공학과 서울대학교 공과대학 제어계측과

A Collision Avoidance Method for Two Articulated Robot Manipulators

C.Chang^o, M.J.Chung
KAIST

B.H.Lee
Seoul National University

Abstract

A simple time-delay method for avoiding collisions between two articulated robot manipulators is proposed. Links of robot are approximated by polyhedra and the danger of collision between two robots is expressed by distances between the robots. An algorithm, which can fast obtain the minimum time-delay value needed for collision avoidance, using scheme of following the boundary contour of collision region in the collision map which has information about collisions between two robots, is described.

I. 서론

다중로봇 시스템은 공통 작업환경 내에서 두대 이상의 로봇이 동시에 작업을 수행하는 응용 시스템으로서 제조작업분야에서 활용이 크게 기대되는 시스템이다. 응용예로는 부피가 큰 물체, 길면서 유연한 작업물의 운반작업, 한대의 로봇으로는 작업이 곤란한 복잡한 조립작업, 그리고 작업을 분담하여 수행하는 병렬작업기능 등을 들 수 있다 [1]. 이 중 병렬작업기능은 단위시간당 작업량을 증가시킬 수 있으므로 대량생산을 목적으로 하는 공장에서 필요한 기능이라 할 수 있다.

그러나 실제로 대부분의 산업현장에서는 다중로봇의 병렬작업기능을 충분히 활용하지 못하고 있다. 이는 여러대의 로봇을 동시에 작업시킬 때 로봇간에 충돌이 일어날 위험이 높기 때문에 작업계획시 충돌회피를 반드시 고려해야 하는데, 아직 다중로봇에서의 충돌문제를 효과적으로 해결할 수 있는 방법이 개발되어 있지 않기 때문이다. 지금까지 소개된 다중로봇을 위한 충돌회피방법들은 [2-6] 각기 나름대로의 장점을 갖고 있으나 적용가능한 로봇형태가 평면적인 운동만이 허락되는 2차원적 로봇이나 로봇의 손목부위만이 고려된 구로봇인 경우와 같이 비교적 단순한 형태로 제한되어 있기 때문에 현재 산업용으로 널리 이용되고 있는 3차원 운동이 가능한 회전다관절형 로봇과 같이 일반적 형태의 로봇에 직접적으로 적용하기는 어렵다.

본 논문은 산업용 로봇의 많은 수를 차지하고 있는 PUMA와 같은 두대의 일반적인 로봇간의 충돌회피를 위한 시간지연방법에 초점을 맞춘다. 시간지연 방법이란 초기에 두 로봇간의 충돌을 고려하지 않고 두 로봇의 시간정보를 기존의 궤도계획 방법에 의하여 구한 다음, 이 초기궤도에서 충돌이 발생할 수 있는 경우 두 로봇중 한대의 로봇이 처음에 예정된 시간보다 늦게 출발함으로써 두 로봇간의 충돌을 피하는 방법을 말한다. 시간지연을 이용한 충돌회피 방법은 로봇으로 하여금 시간지연후 주어진 궤적을 따라 기하학적 경로와 시간정보의 수정없이 움직일 수 있게 하기때문에 용접작업과 패턴링작업과 같이 일정한 경로를 계획된 속도를 유지하며 따라가야 하는 작업에 사용될 수 있는 방법이다.

두 로봇간의 충돌을 피하기 위하여 필요한 최소한의 시간지연값을 구하기 위한 방법으로서 충돌지역 경계선 추적방법이 제안된다. 경계선 추적방법은 두 로봇간의 충돌상황을 효과적으로 나타내는 충돌지도 [4] 상에서 초기에 주어진 TLVST (traveled length versus sampling time) 궤적을 따라 진행하다 충돌지역과 만나게 되면 그 경계선을 따라감으로써 충돌지역을 벗어나고 이 과정에서 얻어진 정보를 이용하여 필요한 시간지연값을 얻는 방법이다. 경계선 추적방법을 사용하게 되면 충돌지도 상에 충돌을 나타내는 충돌지역에 대한 정보를 모두 구할 필요없이 단지 충돌지역 경계선 부근에서의 충돌정보만 구하면 된다. 그러므로, 통상적으로 복잡한 계산과정을 요구하는 전체 충돌지역 구하는 과정이 제외됨으로써 시간지연을 구하는 알고리즘의 구현이 매우 간단해진다. 그리고 충돌회피를 위하여 필요한 시간지연값을 구하는 과정이 특정한 형태의 로봇에 국한되지 않으므로 다양한 형태의 로봇에 대해서 쉽게 구현이 가능하다.

II. 로봇간의 충돌감지

로봇간의 충돌을 감지하기 위해서는 먼저 로봇의 기구학적 구조를 알아야한다. PUMA 로봇은 잘 알려져 있듯이 6개의 자유도를 가진 회전다관절형 로봇으로서 현재 사용되고 있는 로봇중에서 가장 널리 알려진 로봇이다.

PUMA 로봇의 기구학은 D-H (Denavit-Hartenburg) 표현방식에 의해 표현되어진다. 로봇의 링크들은 각각 하나의 다면체 구조로 근사화 되며 로봇의 그립퍼 (gripper) 는 다른 링크들에 비해 복잡한 편이나 여기서는 하나의 다면체로 간단히 모델링한다. 그러므로 한대의 PUMA 로봇은 그림(1)에서 보여지는 것처럼 7개의 다면체 구조들이 연결된 형태로 모델링되어질 수 있다 [7].

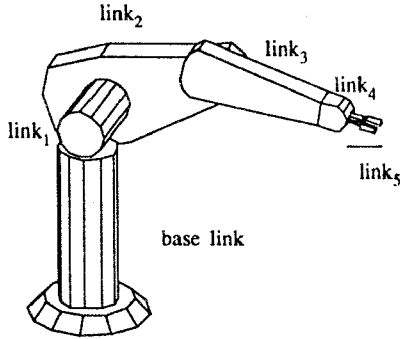


그림 1. 다면체로 모델링된 PUMA-560 로봇.

다면체간의 충돌감지는 다면체간의 거리를 구함으로써 행해질 수 있다 [8]. 즉, 두 다면체간의 최소거리가 영보다 크면 이는 두 다면체는 충돌하지 않았다는 것을 의미하고, 그렇지 않다면 충돌을 의미한다.

m 차원 공간에서 두 다면체 A 와 B 는 두 compact 집합 $K_A, K_B \subset R^m$ 로 각각 표현하면 K_A 와 K_B 의 점들은 두 다면체 A와 B에 의하여 점유되는 공간을 나타낸다. 이때 두 다면체 A와 B 사이의 거리 $d(A, B)$ 는 다음과 같이 정의된다.

$$d(A, B) = \min\{|x - y| : x \in K_A, y \in K_B\} \quad (1)$$

두 로봇간의 충돌감지는 링크들간의 거리를 구함으로써 행해질 수 있다. 두 로봇의 작업환경에서 비록 로봇들이 서로 근접하여 작업하더라도 로봇의 지지대와 첫번째 링크에서 충돌이 일어날 정도로 로봇들을 근접시키는 경우는 극히 드물다. 만일 로봇들을 너무 근접시킨다면 서로간의 간섭이 심하게 되므로 작업효율이 오히려 떨어질 가능성이 크므로 지지대와 첫번째 링크에서는 충돌이 일어나지 않는다고 가정한다. 각 로봇의 지지대 및 첫번째 링크를 제외한 나머지 5개의 링크가 고려되므로 두 로봇간의 충돌감지를 위하여 필요한 거리는 $d(link_1^{(1)}, link_1^{(2)})$, $d(link_1^{(1)}, link_2^{(2)})$, \dots , $d(link_5^{(1)}, link_5^{(2)})$ 모두 25개이다. 편의상 $d(link_i^{(1)}, link_j^{(2)})$ 는 d_{ij} 로 표현된다. 이때 두 로봇간의 거리 $d(R_1, R_2)$ 는 다음과 같이 정의된다.

$$d(R_1, R_2) = \min\{d_{ij}, i, j = 1, \dots, 5\} \quad (2)$$

두 로봇간의 충돌여부는 두 로봇간의 거리를 구함으로써 알 수 있다. 즉, 거리가 영보다 크면 두 로봇간에는 충돌이 일어나지 않음을 뜻하고 그렇지 않다면 충돌임을 의미한다.

III. 충돌지역 경계선 추적에 의한 충돌회피 방법

본 장에서는 두 로봇에 주어진 궤적의 수정없이 초기위치에서의 시간지연을 통하여 충돌을 피하는 방법을 제시하고자 한다. 그림(2)는 로봇-1과 로봇-2의 충돌정보를 로봇-2의 관점에서 나타낸 충돌지도이다. 충돌지도의 가로축은 로봇 제어기의 샘플링타임을 나타내는 s_1 축이다. 샘플링 시간간격을 Δt 라 표기할 때, 샘플링타임 s_1 에 대응하는 실시간 t 는 $t = s_1 \times \Delta t$ 로 주어진다. 충돌지도의 세로축은 로봇-2 손목부위의 이산화된 이동거리를 나타내는 s_2 축이다. 이산화된 이동거리 s_2 에 대응하는 로봇-2의 실제 이동거리 l 은 앞에서 $l = s_2 \times \Delta l$ 이다. Δl 은 샘플링 거리간격을 나타내며, 전체 이동거리 L_T 를 이산화 개수 N_l 로 나눈 값으로 주어진다.

그림(2)에서 TLVST 궤적은 충돌지도 상에서 샘플링타임 축에 대한 로봇-2의 이동거리를 나타낸다. 만일 이 TLVST 궤적이 수정되어진다면, 로봇-2 궤적의 시간정보는 변하게 된다. 그러나, 로봇-1의 시간정보는 변하지 않는다.

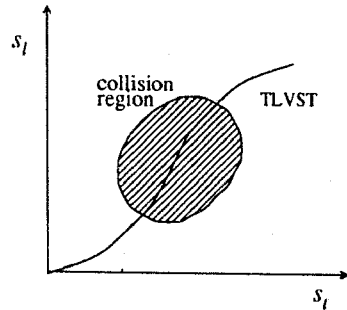


그림 2. 충돌 지도.

충돌지역은 충돌지도 안에서 로봇-1과 로봇-2 사이에 충돌이 발생하는 지역이다. 충돌지도상의 임의의 한점 (s_1, s_2) 에서의 충돌여부는 이점에 대응하는 두 로봇간의 거리가 영보다 크지 아닌지를 검사함으로써 알 수 있다.

초기에 주어진 로봇-2의 TLVST 궤적이 충돌지역을 통과하게 된다면 이는 충돌이 일어난다는 것을 의미하므로 충돌을 피하기 위해서 TLVST 궤적은 충돌지역을 통과하지 않도록 수정되어야 한다. 본 논문에서는 충돌회피를 위한 시간지연을 구할 때 충돌지역에 대한 모든 정보를 구하지 않고 단지 충돌지역의 경계선 부근에서의 충돌정보만을 이용하여 빠른 시간 내에 충돌회피를 위하여 필요한 최소한의 시간지연을 구할 수 있는 경계선 추적방법을 소개하기로 한다.

경계선 추적방법을 이용하여 로봇-2의 시간지연을 구하는 과정을 요약하여 설명하면 다음과 같다 [그림(3) 참조].

- step 1. 주어진 로봇-2의 TLVST 궤적을 따라 진행하면서 충돌여부를 검사해 나간다.
- step 2. 충돌지도 내의 임의의 한 점 (s_i, s_j) 에서 충돌이 일어나면 이탈점 dp를 만날 때까지 충돌지역의 경계선을 따라 진행한다.
- step 3. 이탈점 dp로부터 최종지점까지 TLVST 궤적에 따라 진행한다.

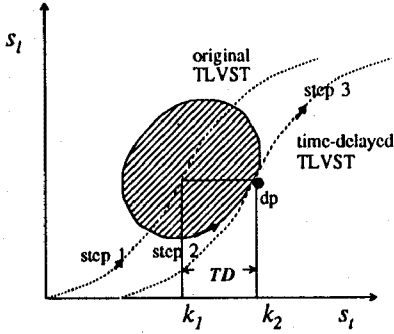


그림 3. 경계선 추적 방식.

이탈점 dp는 충돌지역의 경계선을 따라 진행하는 점들 중에 주어진 TLVST 궤적을 따라 최종위치까지 충돌없이 진행이 허락되는 점을 의미한다. 시간지연 TD는 TLVST 궤적의 이탈점 dp까지의 이동량 $k_2 - k_1$ 과 같다. 진행궤적 (s_i, s_j) 이 충돌지역 경계선을 따라 이동할 때는 진행방향의 왼쪽이 충돌지역의 안쪽이 되도록 한다.

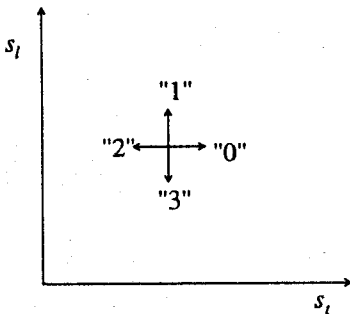


그림 4. (s_i, s_j) 의 4가지 진행방향.

충돌지도 내의 한 점 (s_i, s_j) 에서의 허락되어지는 진행 방향은 그림(4)에서와 같이 "0", "1", "2", "3"등의 3 방향이다. 충돌지도내에서 (s_i, s_j) 의 진행은 TLVST 궤적을 따르는 진행과 충돌지역을 피하기 위하여 충돌지역 경계선을 따라 이동하는 진행등 두가지로 나누어진다. TLVST 궤적은 단조증가의 형태로 주어지므로 TLVST 궤적에 의한 진행은 오직 "0" 또는 "1"의 진행방향들의 시퀀스로 구성된다. 반면에

충돌지역 경계선을 따라 진행할 때에는 "2"와 "3"의 진행방향이 포함될 수 있다. 충돌지역 경계선에서의 임의의 한점 (s_i, s_j) 이 이탈점 "dp"가 될 수 있는 조건은 이 점에서의 진행방향이 "0" 또는 "1"이어야 한다는 것이다. 이탈점의 조건을 만족하는 점에서부터는 다시 기존에 주어진 TLVST 궤적에 따라 진행하게 된다. 지금까지 설명된 과정은 최종위치에 도달할때까지 계속된다. 충돌지역의 경계선을 이루는 점들중에서 이탈점이 될 수 있는 조건을 만족하는 점은 충돌지역의 형태에 따라 여러개 생길 수 있다. 이러한 이탈점의 후보들 중에서 가장 나중에 구해진 점이 최종적으로 이탈점이 된다. 주위할 점은 이탈점의 조건을 만족하여 TLVST궤적을 따라 진행할 때 충돌지역의 형태에 따라 진행이 일정지역에서 계속 반복되어지는 경우가 생길 수 있다. 이러한 현상을 막기 위하여 현재 이탈점의 조건을 만족하는 점이 이전에 이탈점의 후보로 선택되어졌는지 아닌지를 검사하여, 만일 이전에 이탈점의 후보로 선택되어 졌었다면 (s_i, s_j) 의 다음 진행방향을 충돌지역을 따라가도록 정한다. 즉, 새로이 이탈점의 조건을 만족하는 경우에만 TLVST궤적에 따라 진행하도록 한다.

매우 드문 경우라 생각되지만 충돌지도내에 충돌지역이 하나가 아닌 둘 이상이 존재할 가능성도 있다. 이러한 경우를 위하여 로봇-2가 시간지연후 출발부터 이탈점까지 충돌이 발생하는지를 확인할 필요가 있다. 만일 충돌이 발생한다면 충돌지도내에 또하나의 충돌지역이 존재하는 것이므로 이때는 다시한번 충돌지역 경계선 추적을 통하여 새로운 시간지연값을 구해야 한다. 이와같은 과정은 시간지연후 출발에서 이탈점 사이에 충돌이 없을때까지 반복된다.

IV. 경계선 추적방식에 의한 시간지연 방법 적용에

본 장에서는 앞에서 제안된 경계선 추적방식에 의한 시간지연 방법이 어느정도 효과적으로 사용될 수 있는지를 알아보기 위하여 실제로 일어날 수 있는 두 로봇에서의 충돌상황에 대한 시뮬레이션 및 결과에 관해서 논하고자 한다. 그림(5)는 두대의 PUMA-560 로봇, 그리고 로봇 작업대로 구성된 작업환경을 보여준다.

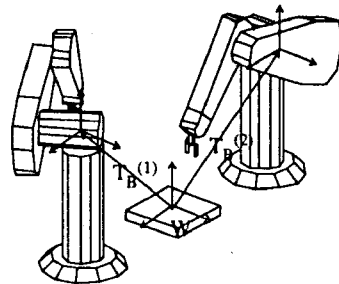


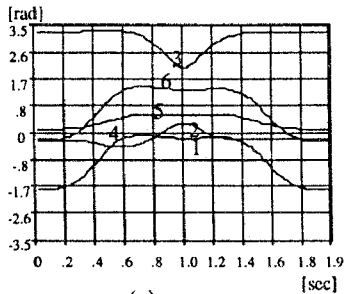
그림 5. 두 PUMA 로봇의 작업환경.

그림(5)에서 W 는 기준좌표계, $T_B^{(1)}$ 과 $T_B^{(2)}$ 는 각각 로봇-1과 로봇-2의 베이스좌표계이다. $T_B^{(1)}$ 과 $T_B^{(2)}$ 는 다음과 같이 주어진다.

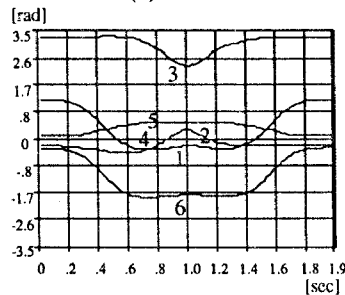
$$T_B^{(1)} = \begin{bmatrix} -1 & 0 & 0 & 0.65 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$$T_B^{(2)} = \begin{bmatrix} 1 & 0 & 0 & -0.65 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

시뮬레이션을 위하여 먼저 로봇들의 초기궤적들이 그림(6)과 같이 주어진다. 이 초기 궤적들은 로봇간의 충돌을 고려하지 않고 계획된 것이다. 또한 초기 궤적들은 로봇의 토크제약조건을 만족한다고 가정한다. 그림(7)은 초기 궤적에 대한 두 로봇간의 거리를 시간축을 따라 보여준다. 시간 0.76[sec]와 1.16[sec] 사이에서 로봇간의 거리가 음의 값을 가지므로 이 시간간격 동안 두 로봇간에는 충돌이 일어나게 된다.



(a)



(b)

그림 6. (a). 로봇-1의 초기 조인트궤적

(b). 로봇-2의 초기 조인트궤적

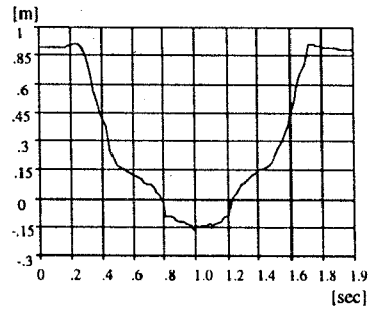


그림 7. 로봇-1과 로봇-2간의 초기 거리.

그림(8)은 앞의 충돌상황에 대한 충돌지도이다. 여기서 샘플링타입은 20ms로 주어지고 s_j 축은 로봇-2의 전체 이동거리에 상관없이 100개의 등간격으로 나누어진다.

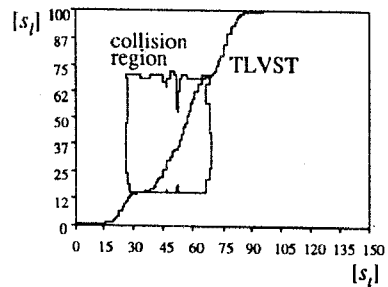


그림 8. 초기 궤적에 대응하는 충돌지도.

그림(9)는 경계선 추적방식에 의하여 충돌회피를 위하여 필요한 시간지연 값을 찾는 과정을 보여준다. 구해진 시간지연은 0.7[sec]이며, 이를 구하기 위하여 소요된 시간은 SUN SPARC 워크스테이션에서 약 90[sec]였다. 그림(10)은 로봇-2가 0.7[sec]만큼 시간지연을 하는 경우에 대한 두 로봇간의 거리 변화를 나타낸다. 이 그림에서 보여지듯이 두 로봇간의 거리는 항상 영보다 크게 된다. 본 장에서 고려된 작업에서는 경계선 추적방식에 의한 시간지연 방법으로 두대의 로봇을 차례로 작업시키는 경우에 비해 1.22[sec]만큼의 시간감축 효과를 얻을 수 있었다.

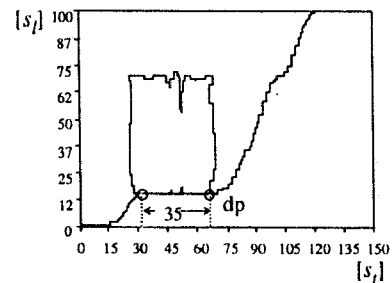


그림 9. 시간지연 알고리즘 수행 결과 :

$$\text{시간지연 TD} = (66-31) \cdot 0.02 = 0.7[\text{sec}]$$

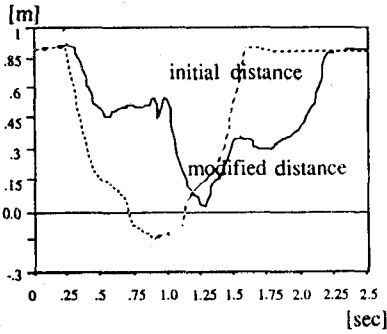


그림 10. 로봇-2의 0.7 [sec] 시간지연후 두 로봇간의 거리.

V. 결 론

공통된 작업환경 내에서 두대의 회전 다관절형 로봇이 충돌없이 안전하게 작업할 수 있도록 하기 위하여 충돌지역 경계선 추적방식에 시간지연 방법을 제안하였다. 본 논문에서 제안된 시간지연 방법을 사용하게되면 일반적으로 많은 계산과정을 필요로 하는 전체 충돌지역 구하는 과정이 제외되므로 알고리즘의 구현 및 계산과정이 매우 간단해지며 로봇의 형태와 모델링 방법에 구애받지 않고 다양한 형태의 로봇에 적용이 가능하게 된다. 경계선 추적방식에 의한 충돌회피 방법은 기존에 주어진 로봇의 기하학적 경로 및 이동속도를 변경하지 않으므로 특정 경로를 반드시 진행해야 하는 아크용접, 도장작업 등과 같은 연속성로 작업에서 특히 유용하다. 물론, 시간지연 방법에 의하여 두 로봇 사이에서 발생할 수 있는 모든 충돌문제를 해결할 수는 없지만 앞에서 언급된 장점들 때문에 두 로봇을 이용하는 많은 작업상황에서 좋은 결과를 얻을 수 있으리라 기대된다.

- [1] R. Mehrotra and M.R. Varanasi, *Multirobot Systems*, IEEE Computer Society Press, 1990.
- [2] K. Kant and S. W. Zucker, "Planning collision-free trajectories in time-varying enviroment: A two-level hierarchy," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 1644-1649, 1988.
- [3] R.S. Wallace, "Three findpath problems," in *Proc. AAAI Nat. Conf. Artificial Intell.*, pp. 326-329, 1984.
- [4] B. H. Lee and C. S. G. Lee, "Collision-free motion planning of two robots," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, no. 1, pp. 21-32, Jan./Feb. 1987.
- [5] 신유식, "두대의 로봇브를 위한 충돌회피 궤적선정," 박사학위 논문, 전기.전자공학, 한국과학기술원, 1989.
- [6] E. Freund and H. Hoyer, "Pathfinding in multi-robot systems: solution and applications," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 103-111, 1986.
- [7] "범용 로봇브 시뮬레이터 개발" 연구보고서, 한국과학기술원, 1989.
- [8] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, "A Fast procedure for computing the distance between complex objects in three-dimensional space," *IEEE J. Robot., Automat.*, vol. 4, no. 2, pp. 193-203, Apr. 1988.