

장 철* 정 명진
한국과학기술원 전기 및 전자공학과

Collision Avoidance Method Using Minimum Distance Functions
for Multi-Robot System

C. Chang* M. J. Chung
KAIST Dept. of Electrical Engineering

Abstract

This paper describes a collision avoidance method for planning safe trajectories for multi-robot system in common work space. Usually objects have been approximated to convex polyhedra in most previous researches, but in case using such the approximation method it is difficult to represent objects analytically in terms of functions and also to describe the relationship between the objects. In this paper, in order to solve such problems a modeling method which approximates objects to cylinder ended by hemispheres and or sphere is used and the minimum distance function is defined which can be calculated simply. Using an objective function with inequality constraints which are related to minimum distance functions, work range and maximum allowable angular velocities of the robots, the collision avoidance for two robots is formulated to a constrained function optimization problem. With a view to solve the problem a penalty function having simple form is defined and used. A simple numerical example involving two PUMA-type robots is described.

I. 서론

로봇의 응용분야 중 조립, 작업을 취급, 용접동과 같은 분야에서는 한 대의 로봇을 사용하는 것 보다 여러 대의 로봇을 동시에 사용함으로써 작업범위를 넓힐 수 있고, 또한 병렬처리로 인한 단위시간당 작업량(through-put)의 향상을 가져올 수 있다. 이러한 다중 로봇 시스템(multi-robot system)의 장점은 특히 대량생산을 목적으로 하는 분야에서 제조원가의 감소 및 생산성의 증가를 꾀할 수 있다.

그러나, 현재 로봇을 사용하고 있는 대부분의 현장에서는 이러한 장점들을 충분히 활용하고 있지 못하고 있는데, 그 이유중의 하나로는 여러 대의 로봇이 공동작업환경(common work-space)내에서 서로 충돌을 하지 않고 작업할 수 있는 효율적인 방법이 거의 없기 때문이다. 그러므로 다중 로봇 시스템을 보다 효과적으로 사용하기 위해서는 서로 충돌치 않고 안전하게 동작시킬 수 있는

체계적인 방법에 대한 연구가 모색되어야 한다.

현재 사용되어지고 있는 많은 로봇들은 여러개의 링크와 조인트로 구성되어 있기 때문에 이들의 기구학적 특성은 연관성이 큰 비선형적이므로 알의 문제를 해결하기란 쉽지 않다. 이를 해결하기 위하여 먼저 수행되어야 하는 것들은 로봇 및 작업물들의 기하학적 모델링과 이들 사이의 상호관계를 기술하는 것이다. 지금까지는 작업물을 convex polyhedra로 근사화 하는 경우가 많았으나 이렇게 모델링하면 작업을 일정한 함수로 표현하기 힘들뿐더러 작업물들의 상호관계식을 정하기가 힘들게 된다. 이로 인하여 충돌문제를 해석적인 방법으로 구체화 하기가 어렵고 해를 얻을 수 있다 하더라도 많은 시간이 걸리게 된다.

본 논문은 이와 같은 문제점들을 해결하기 위하여 작업물들을 양쪽 끝이 반구(hemi-sphere)로 된 원기둥(cylinder)과 구(sphere) 두가지 형태로 나누어 모델링하는 방식을 사용하였다. 이 모델들의 특징은 각 모델을 구성하는 surface가 원기둥에서는 중심을 통과하는 선분에서 구에서는 중심점으로 부터 일정한 거리를 갖고 있는 대칭성 때문에 표현식이 매우 간단해지며 상호관계식도 간단하게 정의될 수 있다. 작업물 간의 관계를 나타내기 위하여 최소거리함수를 정의하였고 이를 구하는 방법에 대해서 기술하였다. 두대의 로봇에 대해 각 링크를 원기둥으로 손목부분을 구로 모델링하고 실행지수(performance index) 및 최소거리함수를 이용하여 비선형 부등구속조건(nonlinear inequality constraint)들을 정의하여 두 대의 로봇에 대한 충돌문제를 구속조건을 갖는 최소화문제로 구체화 하였다. 이 문제에 대한 해를 구하기 위하여 벌칙함수(penalty function)를 사용하는 방법을 소개하였다. 두대의 PUMA 형태의 로봇을 동시에 동작 시키는 과정을 시뮬레이션하고 그 결과들을 보였다.

II. 모델링 및 최소거리함수(Minimum Distance Function)

본 논문에서는 작업물들을 그림 1과 같이 양끝이 반구로 되어 있는 원기둥 및 구 두가지 primitive들을 이용하여 모델링 하였다.

최소거리함수를 이용한 다중 로봇 시스템에서의 충돌회피 방법

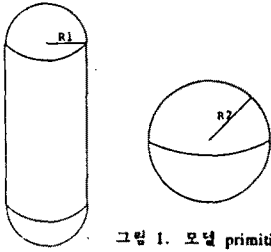


그림 1. 모델 primitive들

이들 두 primitive들을 사용할 경우 필요이상으로 낭비되는 부분이 존재할 수 있지만 실제 작업들의 오차에 의해 발생할 수 있는 문제점들을 피할 수 있고 또한 모델 표현식이 간단해지므로 이들과 관련된 계산들을 빠른 시간안에 할 수 있다.

두 작업들의 surface를 각각 $K_1(x, y, z), K_2(x, y, z)$ 로 놓았을 때 둘 사이의 최소거리함수(MDF)를 다음과 같이 정의한다.

$$MDF = \min[\|z_1 - z_2\| : z_1 \in K_1, z_2 \in K_2] \quad (1)$$

식 (1)에서 z_1, z_2 는 직교좌표계에서의 점들이다.

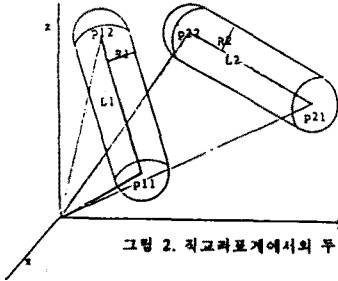


그림 2. 직교좌표계에서의 두 선분

그림 2에서 L_1, L_2 는 각 모델들의 중심을 통과하는 선분들이며 R_1, R_2 는 중심선분에서 surface까지의 최소거리이고 $R_{11}, R_{12}, R_{21}, R_{22}$ 는 각 선분에서의 끝 점들이다. 식 (2)와 같이 주어지는 surface 식들을 가지고 최소거리를 구하기는 어렵다. 그러나 원기둥이 가지고 있는 대칭성을 이용하면 두 surface 사이의 최소거리는 두 선분 사이의 최소거리와 R_1, R_2 를 더한것과 같다는 것을 쉽게 알 수 있으므로 여기서는 3차원 공간상에서 두 선분사이의 최소거리를 구하는 식에 관하여 기술하기로 한다. 먼저 그림 2에서 두 선분 L_1, L_2 를 파라미터 λ_1, λ_2 를 이용하여 식 (2)와 같이 놓는다.

$$\begin{aligned} L_1 &= R_{11} + \lambda_1(R_{12} - R_{11}) \\ L_2 &= R_{21} + \lambda_2(R_{22} - R_{21}) \end{aligned} \quad (2)$$

$$0 < \lambda_1, \lambda_2 < 1$$

L_1 에서의 임의의 점 x_1 과, L_2 에서의 임의의 점 x_2 사이의 거리의 자승 d^2 은 다음과 같이 표현된다.

$$\begin{aligned} d^2 &= [x_{21} + \lambda_2(x_{22} - x_{21}) - x_{11} - \lambda_1(x_{12} - x_{11})]^2 \\ &+ [y_{21} + \lambda_2(y_{22} - y_{21}) - y_{11} - \lambda_1(y_{12} - y_{11})]^2 \\ &+ [z_{21} + \lambda_2(z_{22} - z_{21}) - z_{11} - \lambda_1(z_{12} - z_{11})]^2 \end{aligned} \quad (3)$$

여기서

$$x_1 = (x_{11}, x_{12}, x_{13})^T$$

$$x_2 = (x_{21}, x_{22}, x_{23})^T$$

식 (3)을 바꾸어 표현하면

$$\begin{aligned} d^2 &= (b_1^2 + b_2^2 + b_3^2)\lambda_1^2 + (c_1^2 + c_2^2 + c_3^2)\lambda_2^2 \\ &+ 2(b_1c_1 + b_2c_2 + b_3c_3)\lambda_1\lambda_2 + 2(a_1b_1 + a_2b_2 + a_3b_3)\lambda_1 \\ &+ 2(a_1c_1 + a_2c_2 + a_3c_3)\lambda_2 + (a_1^2 + a_2^2 + a_3^2) \end{aligned} \quad (4)$$

여기서

$$a_1 = x_{21} - x_{11}, a_2 = y_{21} - y_{11}, a_3 = z_{21} - z_{11}$$

$$b_1 = x_{11} - x_{12}, b_2 = y_{11} - y_{12}, b_3 = z_{11} - z_{12}$$

$$c_1 = x_{22} - x_{21}, c_2 = y_{22} - y_{21}, c_3 = z_{22} - z_{21}$$

식 (4)는 다시 최종적으로 식 (5)와 같이 나타낼 수 있다.

$$d^2 = A\lambda_1^2 + B\lambda_2^2 + C\lambda_1\lambda_2 + D\lambda_1 + E\lambda_2 + F \quad (5)$$

여기서

$$A = (b_1^2 + b_2^2 + b_3^2)$$

$$B = (c_1^2 + c_2^2 + c_3^2)$$

$$C = 2(b_1c_1 + b_2c_2 + b_3c_3)$$

$$D = 2(a_1b_1 + a_2b_2 + a_3b_3)$$

$$E = 2(a_1c_1 + a_2c_2 + a_3c_3)$$

$$F = (a_1^2 + a_2^2 + a_3^2)$$

식 (5)의 최소값을 구하기 위해 λ_1, λ_2 에 대해 각각 편미분하여 zero로 놓으면 다음과 같다.

$$\frac{\partial d^2}{\partial \lambda_1} = 2A\lambda_1 + C\lambda_2 + D = 0 \quad (6)$$

$$\frac{\partial d^2}{\partial \lambda_2} = 2B\lambda_2 + C\lambda_1 + E = 0$$

식 (6)에서 알 수 있듯이 d^2 은 (λ_1, λ_2) space의 오직 한 점에서 최소값을 갖게되므로 그 최소값은 만일 식 (6)에서 구해진 λ_1, λ_2 의 값이 0 과 1 사이이면 그 점에 존재하고 그렇지 않으면 최소값은 $0 < \lambda_1, \lambda_2 < 1$ 로 이루어지는 영역의 [그림 3 참조] 경계에 존재하게 된다.

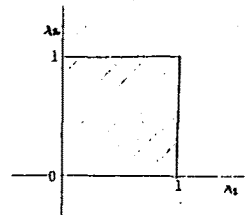


그림 3. (λ_1, λ_2) 영역

최소값이 경계에 존재하는 경우 그 경계를 구성하는 각 edge에서 최소값을 갖는 점들은 다음과 같다.

$$\begin{aligned} \lambda_1 = 0, \lambda_2 &= -\frac{E}{2B} \\ \lambda_1 = 1, \lambda_2 &= -\frac{C+E}{2B} \\ \lambda_2 = 0, \lambda_1 &= -\frac{D}{2A} \\ \lambda_2 = 1, \lambda_1 &= -\frac{C+D}{2A} \end{aligned} \quad (7)$$

그러므로 식 (7)에 주어진 점들에 대응하는 값들 중에서 가장 작은 값이 식 (5)의 최소값이 되며, 점과 점 및 점과 선분 사이의 최소거리함수도 알의 방법과 유사하게 구할 수 있다.

III. 충돌 회피 계획

본 절에서는 공동작업환경 내에 있는 두 대의 로봇들이 각각 수행해야 할 작업과 이들 작업에 관련된 실행지수(performance index)가 주어졌을 때 이를 최소화 하면서 충돌을 피할 수 있는 충돌회피동작계획(collusion-avoidance motion planning)에 관하여 논하기로 한다.

로봇들의 파라미터들과 좌표계는 그림 4와 같이 주어진다.

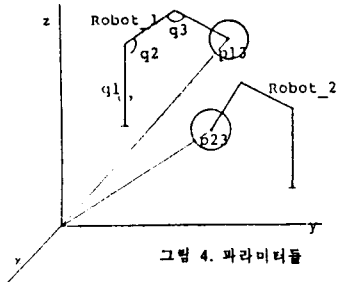


그림 4. 파라미터들

실행지수의 선정은 일의 성격에 따라 달리 정해질 수 있는데 여기서는 직교 좌표계에서 주어진 로봇의 원하여지는 궤도와 실제 궤도 사이의 차이를 고려하여 식 (8)과 같이 정하였다.

$$J = \int_0^T \{ R_{3d}(t) - R_3(q^{(1)}(t), q^{(2)}(t)) \}^T Q \{ R_{3d}(t) - R_3(q^{(1)}(t), q^{(2)}(t)) \} dt, \quad 0 < t < T \quad (8)$$

여기서

$q^{(1)}(t)$: 로봇 1의 처음 3 조인트 벡터

$q^{(2)}(t)$: 로봇 2의 처음 3 조인트 벡터

$R_{3d}(t)$: 로봇 1과 2의 원하는 위치 벡터;
 $(R_{13d}(t), R_{23d}(t))^T$

$R_3(t)$: 로봇 1과 2의 실제 위치 벡터;
 $(R_{13}(t), R_{23}(t))^T$

Q : weighting matrix

그리고 만족해야 할 구속조건들은 다음과 같다.

- 1) 로봇들이 동작시 서로 충돌치 않아야 한다.
- 2) work range를 벗어나지 않아야 한다.

구속조건 1은 앞에서 정의한 최소거리함수를 사용하면 쉽게 구해진다. 먼저 각 링크, 손목사이의 최소거리함수들 즉, $d_1(link_1^{(1)}, link_1^{(2)})$, $d_2(link_1^{(1)}, link_2^{(2)})$, ..., $d_{16}(hand^{(1)}, hand^{(2)})$ 를 이용하여 식 (9)와 같이 표현하면 된다.

$$d_i(t) > C_i \quad i = 1, 2, \dots, 16 \quad 0 < t < T \quad (9)$$

위에 식에서 C_i 는 충돌이 일어나지 않는 안전거리 상수이다. 그리고 work range에 관한 구속조건은 다음과 같이 표현된다.

$$|q_j^{(i)}(t)| < \mu_{ij} \quad i = 1, 2; j = 1, 2, 3 \quad (10)$$

위에 식에서 μ_{ij} 는 로봇 i의 j번째 조인트에서의 최대허용범위이다. 그러므로 두대의 로봇의 충돌문제는 식 (9)와 (10)에 주어진 구속조건을 만족하면서 식 (8)을 최소화 하는 $q^{(1)*}(t)$, $q^{(2)*}(t)$, $t \in [0, T]$ 를 구하는 구속조건이 없는 최소화 문제로 구체화될 수 있다.

일반적으로 위의 문제를 직접 풀기는 어려우므로 여기서는 $[0, T]$ 사이를 N개로 나누어 각 sampling point, 즉 kBr에서 식 (11)와 같이 각 step에서 정의된 실행지수를 최소화 하는 $q^{(1)*}$, $q^{(2)*}$ 를 구한다.

$$\begin{aligned} f = & (R_{3d}(kBr) - R_3(q^{(1)}(kBr), q^{(2)}(kBr)))^T Q \\ & (R_{3d}(kBr) - R_3(q^{(1)}(kBr), q^{(2)}(kBr))) \\ & k = 0, 1, \dots, N \quad (11) \end{aligned}$$

그러나 위와 같은 함수에 대한 최소화 문제를 풀었을 때 $q^{(1)*}(kBr)$, $q^{(2)*}(kBr)$, $k \in [0, N]$, 들이 연속하다는 보장을 못한다. 이를 해결하기 위해서는 식 (12)와 같이 최대 허용속도를

넘지 못하도록 하는 구속조건을 첨가해야 한다.

$$\|(\dot{q}^{(1)}(kBr), \dot{q}^{(2)}(kBr))\| < MAXW \quad (12)$$

여기서

MAXW : 최대 허용 속도

$\dot{q}^{(1)}(kBr)$, $\dot{q}^{(2)}(kBr)$ 를 직접 조인트 변수들의 함수로 나타내기는 힘들므로 다음과 같이 근사화된 식을 이용하기로 한다.

$$\dot{q}_j^{(i)}(kBr) = \frac{q_j^{(i)}(kBr) - q_j^{(i)}((k-1)Br)}{\delta t} \quad (13)$$

$$i = 1, 2 \quad j = 1, 2, 3$$

최소거리함수를 이용한 다중 로봇 시스템에서의 충돌회피 방법

위의 구속조건이 있는 최소와 문제의 해를 구하는 방법에는 대표적인 방법으로 gradient projection 방법 및 벌칙함수를 이용하는 두가지가 있다. 전자는 주로 선형 구속조건들이 주어진 경우에 사용되고 후자의 방법은 비선형 구속조건들이 주어졌을 때 많이 사용된다. 벌칙함수 방법을 사용하여 위의 문제를 식 (14)와 같이 표현하였다.

$$F = f + h_c(q^{(1)}, q^{(2)}) + h_w(q^{(1)}, q^{(2)}) + h_r(q^{(1)}, q^{(2)}) \quad (14)$$

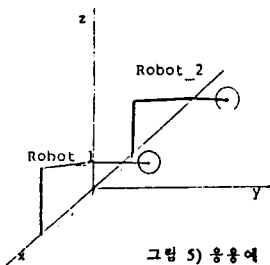
위의 식에서 $h_c(q^{(1)}, q^{(2)})$, $h_w(q^{(1)}, q^{(2)})$, $h_r(q^{(1)}, q^{(2)})$ 는 각각 충돌, 최대회속도, 및 work range에 관한 구속조건들에 대한 벌칙함수이다. 벌칙함수는 여러가지로 정할 수 있으나 본 논문에서는 몇가지의 유형중 아래와 같은 형태로 벌칙함수를 정하였다.

$$h(x) = \alpha \left[\frac{R}{x-C} \right]^m \quad \text{if } x > C \quad (15)$$

윗 식에서 C는 변수 x의 경계치이고 R은 x의 원하는 값에서 대괄호안의 크기가 1이 되도록 주어지는 상수이며, α , m은 임의의 상수이며 식에서 x와 C의 위치는 바뀔 수 있다. 이 함수의 특징은 간단한 형태이면서 원하는 x의 기준점에서 기울기 및 크기들 α , m 을 이용하여 조절할 수 있다는 점이다.

IV. 응용 예

그림 5와 같이 두대의 로봇이 있을 때 앞에서 제안한 방법을 이용하여 충돌회피 동작계획에 대한 시뮬레이션하였다.

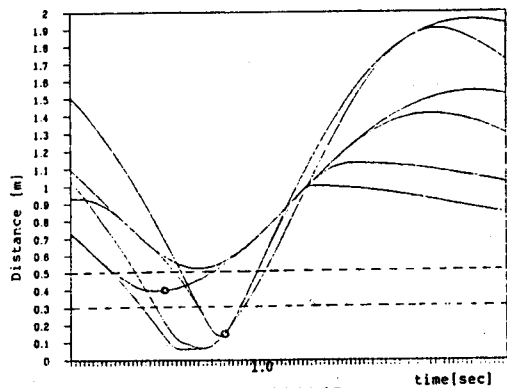
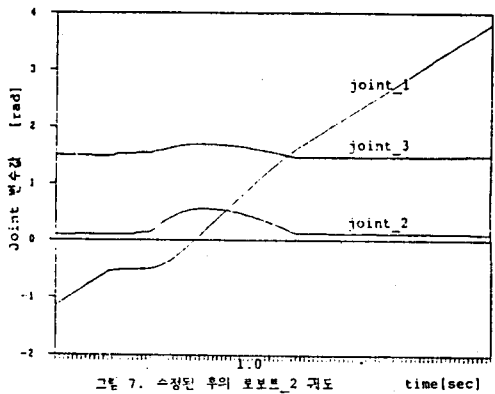
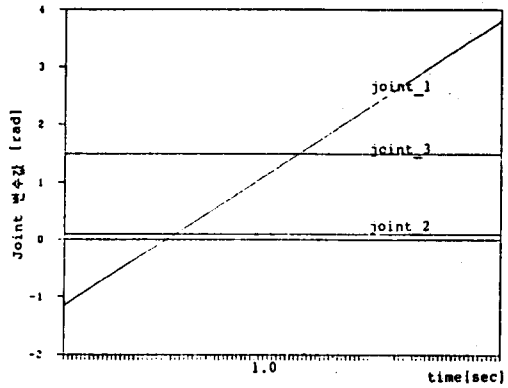


계산의 편리함을 위하여 로봇 1은 초기위치, $q^{(1)} = (0.0, 0.0, 1.57)$ 에 고정시켰으며, 로봇 2는 $q^{(2)} = (-2.0, 0.1, 1.5)$ 에서 $q^{(2)} = (3.0, 0.1, 1.5)$ 로 $q_1^{(2)}$ 이 0.005(rad)만큼씩 증가해 나가도록 하였고 샘플링 시간은 0.01초로 하였다. 식 (14)를 최소화 하기 위하여 급경사(steepest descent)방법을 사용하였다. 식 (14)에서 $h_c(q)$ 의 gradient를 구할 때 어려운 점은 최소거리함수인 d_i 의 미분값을 구하는 것인데 이를 위하여 다음과 같은 근사식을 이용하였다.

$$\frac{\partial d_i}{\partial q} = \frac{d_i(q + \delta q) - d_i(q)}{\delta q} \quad (16)$$

$i = 1, 2, \dots, 16$

그림 (6)은 처음에 주어진 로봇 2의 궤도들이며 (7)은 제한된 충돌회피를 사용하여 수정된 궤도들이고 그림 (8)은 수정되기 전의 몇몇 최소거리함수값들이고 그림 (9)는 수정된 후의 값들이다.



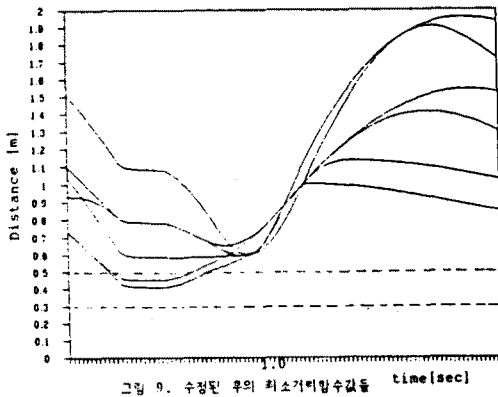


그림 9. 수정된 후의 최소거리함수값들

V. 결론

다중 로봇 시스템에서의 충돌회피 문제를 정의된 최소거리함수를 이용하여 최소화 문제로 구체화 하였고 벌칙함수를 사용하여 해를 구하는 방법을 제안하였으며 시뮬레이션을 통하여 그의 타당성을 보였다. 제안된 방법은 정지물체가 존재하는 경우에도 적용 가능하다. 최소거리함수의 미분시 연속적이지 못한 점들이 나타날 수 있는데 그 영향에 대한 분석 그리고 이를 보상하는 방법에 대한 연구가 필요하다.

참고 문헌

- [1] T. Lozano-Perez and M.A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," Commun. ACM, vol.22, no.10, pp.560-570, Oct.1979.
- [2] T. Lozano-Perez, "Automatic planning of manipulator transfer movements," IEEE Trans. SMC, vol.11, pp.681-698, Oct.1981.
- [3] —, "Spatial planning: A configuration approach," IEEE Trans. Comput., vol.C-32, pp.108-120, Feb. 1983.
- [4] R.A. Brooks, "Solving the find-path problem by good representation of free space," IEEE Trans. SMC, vol. 13, pp. 190-197, Mar. 1983.
- [5] R.T. Cheln et al., "Planning collision-free paths for robotic arm among obstacles," IEEE Trans. PAMI, vol. 6, pp. 91-96, Jan. 1984.
- [6] J.Y.S. Luh and C.E. Campbell, "Minimum distance collision-free path planning for industrial robots with a prismatic joint," IEEE Trans. AC, vol. 29, pp. 675-680, Aug. 1984.
- [7] E.G. Gilbert and D.W. Johnson, "Distance functions and their application to robot path planning in the presence of obstacles," IEEE J. Robot. Auto. , vol. 1, pp.21-30, Mar. 1985.
- [8] B.H. Lee and C.S.G. Lee, "Collision-free motion planning of two robots," IEEE Trans. SMC, vol. 17, no. 1, pp.21-32, Jan./Feb. 1987.