

View Time 개념을 이용한 시변 조인트 제한 지도(JCM)

상에서의 두 로봇의 충돌 회피에 관한 연구

(Time-Varying Joint Constraint Map Using View Time Concept and Its Use on the Collision Avoidance of Two Robots)

南潤錫\*, 李範熙\*, 高明三\*, 高樂溶\*

(Yun Seok Nam, Bum Hee Lee, Myoung Sam Ko, and Nak Yong Ko)

要約

두 로봇이 공동 작업 공간(common workspace)을 구성하고, 그 공간상에서 작업을 할 때 두 로봇은 작업 중 충돌할 수 있다. 본 논문에서는 이러한 충돌 상황을 두 로봇이 동작하기 전에 감지하여 충돌을 회피하는 무충돌 동작 계획 알고리즘(collision-free motion planning algorithm)을 view time 개념을 이용하여 제안한다.

두 로봇간에는 엔드 이펙터(end-effector)뿐만 아니라 링크의 충돌이 발생할 수 있다. 충돌 감지와 회피를 위하여 로봇 1의 궤적을 주기적으로 view time마다 샘플링(sampling)하여, 샘플링 구간 동안 발생하는 그 로봇의 자취 영역(swept region)을 장애물로 모델링하였다. 이 “장애물”에 대한 로봇 2의 조인트 제한 영역을 구하고, 이를 조인트 제한 지도(joint constraint map:JCM)로서 각 view time마다 조인트 공간상에 표현하였다. 이 지도들간의 변화를 고려하여 두 로봇 시스템의 무충돌 동작을 계획하는 알고리즘을 구성하였으며, 이를 검증에 의하여 확인하였다.

Abstract

Two robots working in a common workspace may collide with each other. In this paper, a collision-free motion planning algorithm using view time concept is proposed to detect and avoid collision before robot motion. Collision may occur not only at the robot end effector but also at robot links. To detect and avoid potential collisions, the trajectory of the first robot is sampled periodically at every view time and the region in Cartesian space swept by the first robot is viewed as an obstacle during a single sampling period. The forbidden region in the joint space for the second robot is obtained from this “obstacle,” and represented by the joint constraint map (JCM). The JCM's are obtained in this way at every view time. An algorithm is established for collision-free motion planning of the two robot system from the sequence of JCM's and it is verified by simulations.

I. 서론

로봇의 다른 물체에 대한 충돌은 그 상황이 돌발적이고 작업자의 안전 문제, 작업 물체의 파손 등 작업 환경에 큰 영향이 있으므로 이에 대한 대책이 필요하게 되었다. 작업 공간내에 위치한 물체의 충돌 회피에서 먼저 고려되어야 할 것은 물체의 존재 여

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

接受日字: 1989年 6月 15日

(※ 본 연구는 한국과학재단 신진연구비(883-0803-007-2)의 지원으로 수행되었음.)

부 결정이고, 존재할 경우에는 그 물체의 위치, 형태, 방향등의 기하학적 특성과 물체의 유동 여부에 관한 정보와 유동 물체인 경우에는 그 물체의 경로(path)와 궤적(trajjectory)에 대한 정보가 필요하다. 로봇트는 이러한 정보를 자신의 경로 및 궤적과 비교하여 충돌 가능성을 판단하는데 이를 충돌 감지(collision detection)라 한다. 또한, 충돌 가능성이 있는 경우에는 충돌 위치와 시간등을 구하고, 이를 이용한 충돌 회피 알고리즘을 수행하여 새로운 경로의 궤적을 구성한다. 이 일련의 과정을 무충돌 동작 계획(collision-free motion planning)이라 한다. 무충돌 동작 계획은 무충돌 경로 계획과 무충돌 궤적 계획으로 구분할 수 있고,<sup>10)</sup> 무충돌 동작 계획은 고정 물체(stationary object)<sup>11-12)</sup> 인가 유동 물체(moving object)인가에 따라 그 방법 상에 차이가 있다. 유동 물체의 회피는 그 물체의 기하학적인 정보뿐 아니라 궤적등의 운동 정보가 필요하고, 고정 물체에 비해 시간을 고려해야 하므로 문제가 더욱 복잡해진다. 이러한 유동 물체의 회피 문제를 해결하기 위한 여러 연구가 이루어지고 있다.<sup>8-12)</sup>

유동 물체의 충돌 회피에 관한 연구는 대략 3가지 유형으로 나눌 수 있다. 첫 번째 유형으로는 속도 변경법을 사용한 유동 장애물들의 충돌 회피 동작을 하는 방법이다. Lee<sup>10)</sup>는 물체의 속도를 고려하여 작성된 충돌지도(collision map)를 기반으로 로봇트의 동작을 시간 스케줄링(time scheduling)하여, 구(sphere)로 모델링된 로봇트의 팔목(wrist)의 유동 환경 내에서의 충돌 회피에 관한 알고리즘을 제안하였다. Kant<sup>11)</sup>는 경로 계획과 속도 계획 문제를 분리하여 무충돌 동작 계획을 수행하는 알고리즘을 제시하였다. 두 번째 유형으로는 로봇트가 다른 유동 물체에 대해 충돌 가능한 형상(configuration) 범위를 단위 샘플링 시간(sampling time)마다 구성하는 것이다. 이렇게 작성한 형상 공간-시간 지도(configuration space-time map)를 이용하여 무충돌 동작 계획을 하는 방법이 제안되었다.<sup>8)</sup> 한편, Park는 형상 공간 개념을 이용하여 다중 로봇트 시스템의 동작 계획기(motion planner)를 구성하였다.<sup>9)</sup> 세 번째 유형으로 Freund<sup>12)</sup>는 계층 코오디네이션 제어(hierarchical coordination control)를 이용하여 다중 로봇트 시스템의 동작을 온라인(on-line)으로 제어하는 알고리즘을 제시하였다.

본 논문은 2대의 로봇트가 공동 작업 공간을 가지고 작업하는 경우에 있어 종래의 연구와는 다른 방법으로 두 로봇트가 충돌없이 주어진 동작을 수행하게 하는데 목적이 있다. 한 로봇트(R1)가 주어진 경로와 궤적으로 목적지까지 움직일때, 다른 로봇트(R2)

와의 충돌 여부를 결정하고, 충돌시 회피 동작을 수행해야 한다. 이때, 로봇트 R2가 보면 로봇트 R1이 계속 움직이므로 각 순간마다 두 로봇트의 비교는 시간상 어려운 일이다. 이런 문제를 해결하기 위하여 본 논문에서는 일정한 시간(view time)마다 로봇트 R1의 동작을 샘플링하여 샘플링 구간 동안에 로봇트 R1이 움직인 자취 영역(swept region : SR)을 구한다. 이것은 이 영역을 그 구간에서의 회피해야 할 고정된 장애물로 보면 문제 해결이 용이하기 때문이다. 또한, 각 샘플링 구간마다 로봇트 R2가 로봇트 R1의 움직임으로 인하여 제한 받는 기하학적 시변 조인트 제한(time-varying joint constraint)을 조인트 공간 상에 표시한 조인트 제한 지도(joint constraint map:JCM)를 작성한다. 이것은 직각 좌표계 상에서는 로봇트 몸체 전체에 대한 회피 동작 계획은 어려우나, 조인트 공간상에서는 마치 직각 좌표계 상의 한 점(point)이 임의의 물체를 회피하는 것과 같이 회피 동작을 수행할 수 있기 때문에 종래의 회피 동작 알고리즘을 이용할 수 있다는 장점이 있다. 본 논문에서는 이렇게 구성된 시변 JCM을 이용하여 조인트 공간에서의 무충돌 동작계획 알고리즘을 제안하였다.

## II. 두 로봇트 시스템의 구성

공동 작업 공간(Common Workspace:CW)은 여러 로봇트가 컨베이어 벨트나 작업대를 두고 공동 작업을 하는 작업 공간으로 정의할 수 있는, 로봇트간의 충돌 가능 지역이다. 그림 1은 두 로봇트 시스템과 공동 작업 공간(A 지역)을 나타낸 그림이다. 본 논문에서 고려하는 시스템은 다음과 같은 특성을 갖는다.

(1) 로봇트는 회전 조인트(rotary joint)로 된 2 자유도의 평면형이다. 로봇트 링크는 직사각형이고, 링크의 길이는 각각  $a_1$  (350mm),  $a_2$  (250mm) 이고, 너비는  $r$  (25mm)이다.

(2) 회전 범위 :  $\theta 1 \min(-130^\circ) \leftrightarrow \theta 1 \max(100^\circ)$   
 $\theta 2 \min(-120^\circ) \leftrightarrow \theta 2 \max(120^\circ)$

최대 속도 :  $|\dot{\theta}1 \max| = 131$ ,  $|\dot{\theta}2 \max| = 180$   
 (deg/sec)

최대 가속도 :  $|\ddot{\theta}1 \max| = 656$ ,  $|\ddot{\theta}2 \max| = 900$   
 (deg/sec<sup>2</sup>)

(3) 두 로봇트 좌표계의 X축은 동일 직선상에 있고, 공동 작업 공간을 갖는다. 즉,  $a_1 + a_2 > dx/2$ 이다. (dx는 두 로봇트 사이의 거리)

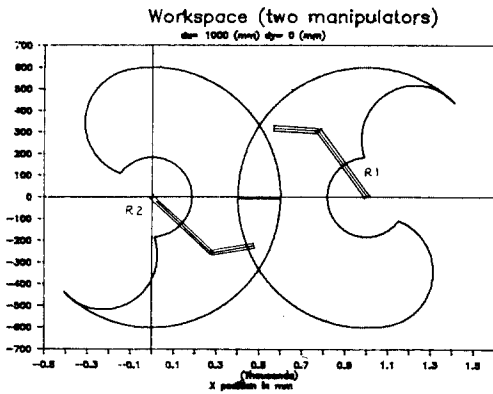


그림 1. 두 로봇 시스템  
Fig. 1. The two-robot system.

(4) 로봇 R1은 로봇 R2에 대하여 동작의 우선권을 갖고, 로봇 R2는 동작 전에 로봇 R1의 제적과 기하학적 정보를 알 수 있다.

(5) 초기에 주어진 로봇 R1의 경로는 직각 좌표계에서 직선이고, 로봇 R2의 경로는 조인트 좌표계에서 직선이다. 그러나, 로봇 R2의 무충돌 동작 계획은 주어진 도달 시간이나 경로등을 변경할 수 있다. 또한, 로봇 동작중 팔 형상(arm configuration)은 일정하다.

### III. 조인트 제한 지도의 구성

로봇 몸체 전체에 대한 충돌 회피 문제를 쉽게 다루기 위한 방법으로 직각 좌표계 상의 물체를 조인트 좌표계 상으로 사상(mapping)하는 방법을 사용한다. 이 방법은 조인트 좌표계 상에서의 로봇을 한 점(point)으로 보고 회피 동작 계획을 할 수 있는 점 이외에도 여러가지 이점이 있다. 로봇 R1에 대한 로봇 R2의 충돌 가능 조인트 영역을 구하기 위해서 다각형-다각형 접촉 판명법<sup>1)</sup>을 이용한다. 그러나, 로봇 R1이 움직이므로 이 영역을 구하기는 쉽지 않다. 이러한 문제를 해결하기 위하여 로봇 R1의 제적을 view time 마다 샘플링하여 각 샘플링 구간(view period)동안에 로봇 몸체가 거처간 직각 좌표계 상에서의 영역을 볼록 다각형(convex polygon)으로 근사시킨 SR을 구성한다. 각 view time에서의 SR은 정지 물체로 볼 수 있기 때문에 다각형-다각형 접촉 판명법의 적용이 가능하다.

#### 1. 시변 장애물의 구성과 모델링

시변 물체(time-varying object)란 시간이 흐름에

따라 위치나 방향등이 변화할 뿐 아니라, 형태도 변화하는 물체로 정의 된다.<sup>11)</sup> 본 논문에서는 로봇의 두 링크들의 실제 자취 영역을 포함하는 가장 작은 볼록 다각형을 SR로 정의한다. 시변 물체에 대한 SR도 시간의 흐름에 따라 위치, 방향, 형태가 달라지므로 역시 시변 물체이다. 로봇 핸드와 직선 운동을 하므로 로봇의 현재 형상과 다음 view time에서의 형상을 포함하는 가장 작은 볼록 다각형을 구하면 SR이 된다. 그런데, SR을 구하는 알고리즘은 로봇의 현재 위치(S)에 대하여 다음 view time 시의 위치(G)가 그림 2의 영역 A, B, C, D 중 어디에 속해 있는가에 따라, 그리고 로봇 형상에 따라 달라진다.<sup>12)</sup> 그림 2는 G가 영역 B에 있는 경우의 SR을 도시하였다. 다음은 영역 B에 대한 SR 구성 방법인데 그림 2의 SR은(경우 1)에 해당된다.

#### ●영역 B에 대한 SR 모델링(좌측 팔 형상인 경우)

(경우 1)  $(\theta_2)_i < 90^\circ$  일때

$$SR : 0 - (J)_i - S - G - 0$$

(경우 2)  $(\theta_2)_i \ge 90^\circ$  일때

(경우 2, 1)  $0 \overline{(J)_f}$  와  $\overline{(J)_i S}$  사이에 교점이 있고,

$\overline{(J)_f G}$  와  $\overline{(J)_i S}$  사이에도 교점이 있을때

$$SR : 0 - (J)_i - CP - (J)_f - S - G - 0$$

(경우 2, 2)  $0 \overline{(J)_f}$  와  $\overline{(J)_i S}$  사이에 교점이 있을때

$$SR : 0 - (J)_i - CP - (J)_f - G - 0$$

(경우 2, 3)  $\overline{SG}$  와  $0 \overline{(J)_f}$  사이에 교점이 있을때

$$SR : 0 - (J)_i - CP - (J)_f - G - 0$$

(경우 2, 4) 교점이 없을때

$$SR : 0 - (J)_i - CP - S - G - 0$$

(사용 기호)

$\overline{MN}$  : 점 M과 N을 잇는 선분, 0 : BASE

$(J)_i, (J)_f$  : 직각 좌표계상의 조인트 2의 현재 및 목표 위치

$(\theta_2)_i$  : 조인트 2의 현재 회전각

그림 3은 직선으로 움직이는 로봇 운동을 네 부분으로 분할하여 각 view time 마다 구성된 SR을 도시하였다. 이 SR은 시간이 지남에 따라 형태, 위치등이 변함을 알 수 있다. 이러한 SR은 선분 링크로 된 로봇에 대한 것이나, 직사각형 링크이므로 모델링한 SR의 각 모서리를 r 만큼 증가시켜야 한다. 우측 팔 형상인 경우도 구성 원리는 좌측 팔 형상과 같다.

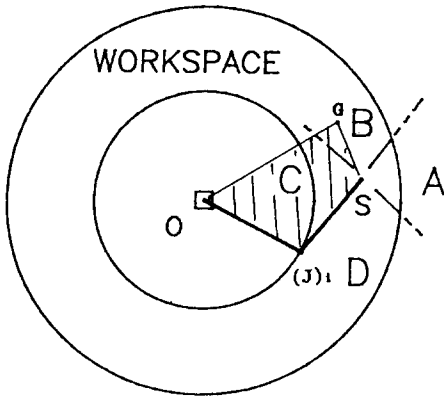


그림 2. SR의 모델링 영역과 SR 구성 예(B 구역)  
 Fig. 2. The regions for SR construction and an example of SR(Region B).

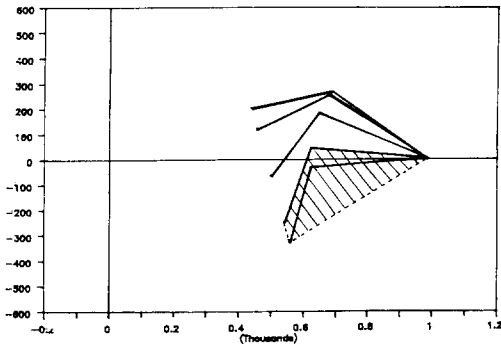


그림 3. 직선 운동에 의해 발생하는 SR(총 동작 시간을 네 부분으로 나눔)  
 Fig. 3. The SR generated by straight line motion (4 sampling times).

2. 조인트 제한 지도의 구성 알고리즘

임의의 위치에 있는 SR과 로봇 링크가 접촉하는 로봇 조인트 값의 범위를 구하는 방법으로서 다각형-다각형 접촉 판별법을 사용한다.<sup>1)</sup> 이 방법은 임의의 볼록 다각형의 모서리에 대한 다른 볼록 다각형의 꼭지점 벡터가 회전하여 접촉하게 되는 각도의 범위를 구하는 것이다. 이러한 방법으로 각 링크에 대해 SR 모서리-링크 꼭지점, SR 꼭지점-링크 모서리간의 접촉 회전각 범위를 구한다. 두 경우가 같은 방법이므로 SR 모서리-링크 꼭지점 비교에 대해서 서술한다.

구하고자 하는 것은 그림 4에서 임의의 벡터  $\nabla$

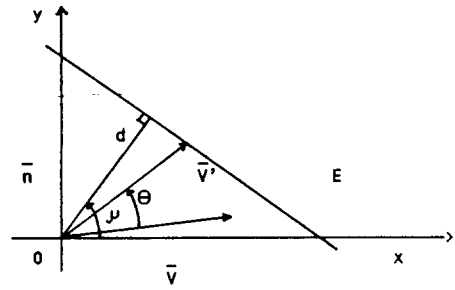


그림 4. 모서리에 대한 꼭지점의 접촉각 구하는 방법  
 Fig. 4. The contact angle of the vertex against the edge.

가 직선 E와 접촉하는 벡터  $\nabla$ 와  $\nabla'$ 의 회전 각도  $\theta$ 이다. 직선 E는 식 (1)과 같이 표현할 수 있다. 여기서, d는 좌표계 원점에서 이 직선까지의 수직 거리이고,  $\mu$ 는 X축에 대한 수직선의 각도이다.

$$X \cos \mu + Y \sin \mu = d \tag{1}$$

벡터  $\nabla'$ 는 식 (2)와 같고, 이 식을 식 (1)에 대입하여  $\theta$ 에 대해서 정리한 것이 식 (3)이다.

$$\begin{bmatrix} V_x' \\ V_y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} V_x \\ V_y \end{bmatrix} \tag{2}$$

$$A \cos \theta + B \sin \theta = d \tag{3}$$

여기서,  $A = V_x \cos \mu + V_y \sin \mu$   
 $B = V_x \sin \mu - V_y \cos \mu$ 이다.

식 (3)으로 부터  $\theta$ 값을 구하면 식 (4)와 같다.

$$\theta = \cos^{-1} \frac{A \times d \pm \sqrt{(A \times d)^2 - (A^2 + B^2) \times (d - B)^2}}{(A^2 + B^2)} \tag{4}$$

식 (4)에서 4개의  $\theta$ 값이 발생한다. 이중  $|\cos \theta| \leq 1$ 이 되는 2개의  $\theta$ 값만을 취한다. 한편, SR의 꼭지점-링크의 모서리 비교시에는 식 (4)와 반대 부호의 식을 얻을 수 있다. 이것은 꼭지점의 회전 방향이 로봇 꼭지점-SR 모서리 경우와 반대(시계 방향)이기 때문이다. 여기서 구한  $\theta$ 값들이 원하는 값이 되기 위해 다음 2가지 제한 조건을 만족해야 한다.

● 제한 1 (In-edge constraint)

벡터  $\nabla$ 를  $\theta$ 만큼 회전한 벡터  $\nabla'$ 가 모서리 위에 있는지 검증해야 한다. 그림 5에서 X축으로부터

$\theta_i$  회전한 벡터  $\vec{V}_i$ 는 모서리 위에 있지 않고,  $\theta_j$  회전한 벡터  $\vec{V}_j$ 는 모서리 위에 있다. 이것은 모서리의 선분만을 고려한 것이 아니라 직선을 고려하여 충돌각을 구했기 때문이다. 이 경우에는 계산을 통해 모서리위의 값만을 취한다.

● 제한 2 (orientation constraint)

이 제한은 그림 6의 점선 형상과 같은 경우를 제거하기 위한 것이다. 이런 경우가 발생하는 것은 비교 대상이 선이나 점이나 다각형이기 때문이다. 식 (4)의  $\theta$ 값이 그림 6의 실선 형상의 조건을 표시한 식 (5)의 조건을 만족하면 원하는 충돌각을 얻게 된다.

$$\text{Sgn}(\bar{n} \cdot \bar{e}_1) \geq 0 \quad \text{그리고} \quad \text{Sgn}(\bar{n} \cdot \bar{e}_2) \geq 0 \quad (5)$$

단,  $\bar{m} \cdot \bar{n}$ 은 벡터  $\bar{m}$ 과  $\bar{n}$ 의 내적(inner product)이다 위의 방법으로 SR의 모든 모서리에 대해 충돌값을 구한다. 한 링크에 대한 SR의 비교시에는 비교 대상이 불록 다각형이기 때문에 다각형의 꼭지점을 제외하고는 2개의 접촉각을 구할 수 있다. 이것이 그 view period에서의 충돌 가능 상한값과 하한값이다. 2링크 로봇트 경우에 대해서 과정을 요약하면 다음과 같다.

(단계 1) 조인트 1을 원점으로 하여 SR의 한 모서리에 대한 링크 1의 각 꼭지점의 충돌 가능 조인트 영역을 구하고, 각 모서리에 대한 구한 값들의 합집합( $R(\theta_1)$ )을 구한다. 반대의 경우에 대해서도 제한 영역을 구한다.

(단계 2)  $R(\theta_1)$ 을 제외한 영역에 대해 조인트 2를 원점으로 하여 링크 2에 대하여 단계 1과 같은 과정을 수행한다.

위의 과정을 통해 구한  $\theta_1, \theta_2$ 의 제한 영역을 조인트 좌표계상에 표시한 것이 JCM이다.

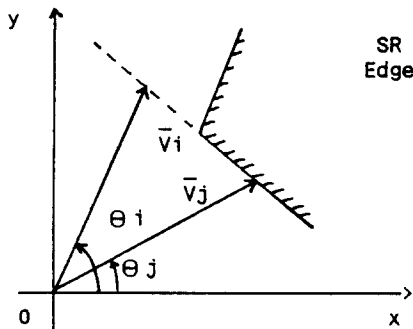


그림 5. 접촉점이 In-edge 제한에 저촉되는 경우 Fig. 5. The in-edge constraint.

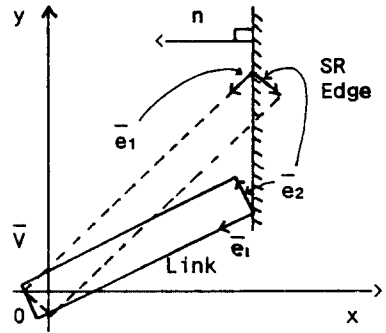


그림 6. 방향 제한 Fig. 6. The orientation constraint.

IV. 무충돌 동작 계획

JCM 상에서의 조인트 제한 영역(이후 Forbidden Region:FR)은 조인트 좌표계 상의 장애물을 뜻한다. JCM 상에는 로봇트 R2의 시점, 종점과 궤적을 표시할 수가 있는데, 이것은 마치 직각 좌표계 상에서 고정 물체를 회피하는 것과 같이 JCM 상에서 FR을 회피하여 목적 형상(JCM 상의 한점)까지 가는 것과 같다. 본 논문은 이를 위해 V-그래프 탐색 방법(V-graph search)을 이용하는 알고리즘을 제안한다. 그런데, 이 탐색법을 이용하는데 있어 두 가지의 문제점이 있다. FR이 불규칙 물체라는 점과 각 view time 마다 작성된 JCM(이후 슬라이스)상의 FR이 시변 물체라는 점때문에 이 방법을 적용하는데 어려움이 있다. 첫 문제는 그림 7과 같이 FR을 직사각형으로 근사화함에 의해, 두 번째 문제는 로봇트 R2의 궤적 계획을 함에 있어 현재 슬라이스와 다음 슬라이스를 모두 고려함으로써 해결할 수 있다.

1. JCM 상에서의 무충돌 경로 계획과 궤적 계획 임의의 k-슬라이스(k-th view time~(k+1)-th view time:k=1..n)에서 시점( $P_s(k)$ )과 종점( $P_g(k)$ )의 FR에 대한 상대적 위치에 따라 여러 형태의 경로가 발생할 수 있다. (그림 7) 본 논문에서는 V-그래프 탐색법을 사용하여 한 슬라이스에서의 최단 경로를 구한다.

전가속(full acceleration), 전감속(full deceleration)의 포물선과 선형함수의 혼합방식(linear function with parabolic blends)의 궤적 계획법을 이용한다. 구하고자 하는 것에 따라 두 가지의 궤적 계획방법으로 구분할 수 있다. 한 가지는 도달 거리를 알고 도달 시간을 구하고자 할 때이고(경우 1), 다른 경우는 도

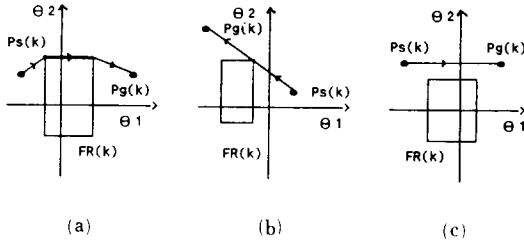


그림 7. k-슬라이스에서 구한 최단 경로 예  
Fig. 7. The shortest path in the k-th time slice.

도달 시간을 알고, 도달 거리를 알고자 하는 경우이다 (경우 2).

(경우 1) 도달 거리는 알고, 도달 시간을 구하고자 할 경우.

(경우 1.1)  $dis < d(2 \times t_b)$  일때 (그림 8 (a))

$$t_f = \sqrt{(4 \times dis) / \dot{\theta} \max}$$

(경우 1.2)  $dis \geq d(2 \times t_b)$  일때 (그림 8 (b))

$$t_f = (dis / \dot{\theta} \max) + t_b$$

(경우 2) 도달 시간은 알고, 도달 거리를 구하고자 할 경우.

(경우 2.1)  $t_f < (2 \times t_b)$  일때 (그림 8 (a))

$$dis = (1/4) \times t_f^2 \times \ddot{\theta} \max$$

(경우 2.2)  $t_f \geq (2 \times t_b)$  일때 (그림 8 (b))

$$dis = (t_f - t_b) \times \dot{\theta} \max$$

(사용 기호) 도달 거리 : dis, 도달시간 :  $t_f$ ,  
가속(감속)시간 :  $t_b = \dot{\theta} \max / \ddot{\theta} \max$ ,  
가속(감속)거리 :  $d(t_b) = t_b \times \dot{\theta} \max$ ,  
 $\dot{\theta} \max$ ,  $\ddot{\theta} \max$  : 최대 속도, 가속도

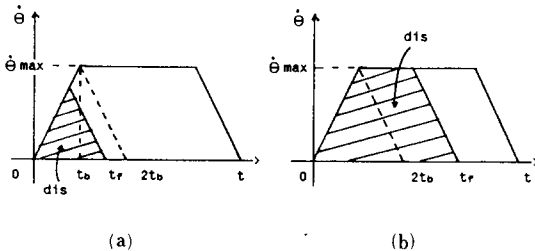


그림 8. 전 가속, 전 감속의 포물선과 선형 함수의 혼합 방식의 궤적 계획  
Fig. 8. The linear function with parabolic blends trajectory planning method using full acceleration/deceleration.

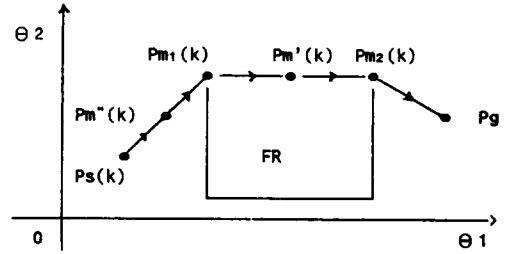


그림 9. k-슬라이스 내에서 발생 가능한 경유 점들  
Fig. 9. The via points occurring in the k-th time slice.

그림 9와 같이 k-슬라이스에서 경로  $(P_s(k) \rightarrow P_{m_1}(k) \rightarrow P_{m_2}(k) \rightarrow P_g)$ 를 따라 궤적 계획을 할 경우 로봇의 속도와 가속도의 제한 때문에 k-슬라이스 내에서  $P_g$  까지 가지 못하는 경우가 생긴다. 이 경우  $P_s(k)$ 와  $P_g$  사이의 경로 상에서, k-슬라이스의 마지막 순간에 도달하는 점을 k-슬라이스에서의 경유점이라고 정의한다. 경유점의 위치는 그림 9에서와 같이 경로  $(P_s(k) \rightarrow P_{m_1}(k))$  사이 또는 경로  $(P_{m_1}(k) \rightarrow P_{m_2}(k))$  사이에 위치할 수 있다. 각 경유점이 발생하는 경우와 경유점을 구하는 방법은 다음과 같다.

(경우 A)  $P_{m'}(k)$ 가 경유점이 되는 경우(식(6))

$P_{m_1}(k)$ 까지 걸리는 시간( $T_m = \text{MAX}(T_{m_1}, T_{m_2})$ ):  $T_{m_i}$ 는 조인트 i의 동작 시간을 (경우 1)의 궤적 계획법을 이용하여 구한다.

k-슬라이스 내에서 조인트 i가 동작할 수 있는 최대 회전각을  $\theta_{i \max}$ 라하면,  $\theta_{i \max}$ 가 그림 9의  $P_{m_1}(k)$  점에서의  $\theta_i$ 보다 크면 경유점은  $P_{m'}(k)$ 와 같은 곳에 위치하게 된다(식(6)).

$$|\theta_i(P_{m_1}(k)) - \theta_i(P_s(k))| \leq \theta_{i \max}, i = 1, 2 \quad (6)$$

이런 경우 (경우 1)의 궤적 계획법으로  $P_{m_1}(k)$  점까지 운동하는 데 소요되는 시간을 구하고,  $P_{m_1}(k)$ 에서부터 남은 시간동안 ( $V_t - T_m : V_t$ 는 view time)에 도달할 수 있는 점을 구하면  $P_{m'}(k)$ 를 구하게 된다.

(경우 B)  $P_{m''}(k)$ 가 경유점이 되는 경우

$\theta_{i \max}$ 가 그림 9의  $P_{m_1}(k)$  점에서의  $\theta_i$ 보다 작으면 k-슬라이스 내에서 도달할 수 있는 점은  $P_{m_1}(k)$  이내에 있게되어 경유점은  $P_{m''}(k)$ 에 위치하게 된다. 즉, 식(7)또는(8)의 경우에 경유점이  $P_{m''}(k)$ 에 위치하게 된다.

$$|\theta_1(P_{m_1}(k)) - \theta_1(P_s(k))| > \theta_1 \max \quad (7)$$

$$|\theta_2(P_{m_1}(k)) - \theta_2(P_s(k))| > \theta_2 \max \quad (8)$$

식 (7)의 의 경우에는  $\theta_1$ 에 대해서 (경우 2)의 방법을 적용하고 식 (8)의 경우에는  $\theta_2$ 에 대해서 (경우 2)의 방법을 적용하여 경유점  $P_{m''}(k)$ 를 구한다.

2. 경유점과 다음 슬라이스와의 충돌 비교

한 슬라이스에서 구한 경유점이 다음 슬라이스 상의 FR의 내부에 있을 때는 직각 좌표계 상에서 로봇 사이에 충돌이 없어도 다음 슬라이스에서는 충돌 가능성이 있다는 것을 뜻한다. 이런 경우에는 두 슬라이스의 FR을 모두 회피하는 경로, 즉 경유점을 새로이 구해야 한다. 현재 슬라이스(k-slice)상의 FR은  $FR(k)$ , 다음 슬라이스((k+1)-slice)의 FR은  $FR(k+1)$ 이라 하고, 현재 슬라이스의 시점  $P_s(k)$ 에 대하여 k-슬라이스에서 구한 경유점은  $P_{m'}(k)$ , k-슬라이스에서 (k+1)-슬라이스 FR에 대하여 구한 경유점은  $P_{m'}(k+1)$ 라 한다. 현재 슬라이스에서 구한 경유점과 다음 슬라이스의 FR을 비교할 때 다음의 발생 가능한 세 가지 경우가 있는데, 각 경우와 해결 방법은 다음과 같다.

- $P_{m'}(k) \in FR(k+1)$ 인 경우(그림 10(a)) :

$FR(k+1)$ 에 대한 새로운 경로와 궤적 계획으로  $P_{m'}(k+1)$ 을 구한다.

- $P_{m'}(k) \in FR(k+1)$ ,  $P_s(k) \notin FR(k+1)$ 이고,

$(P_s(k) P_{m'}(k+1)) \cap FR(k) \neq \emptyset$ 인 경우(그림 10(b)) :  $FR(k)$ 과  $FR(k+1)$ 을 모두 회피하는 경로와 경유점( $P_{m''}(k+1)$ )을 구한다.

- $P_{m'}(k) \in FR(k+1)$ ,  $P_s(k) \in FR(k+1)$ 이고,

$P_{m'}(k+1) \in FR(k+1)$ 인 경우(그림 10(c)) :

$P_s(k)$ 에서부터  $FR(k+1)$ 의 가장 가까운 꼭지점이나 모서리 중의 한점을 경유점( $P_{m''}(k+1)$ )으로 한다.

단,  $FR(\cdot)$ 은 경계선을 제외한 직사각형 내의 점들의 집합이다.

3. 무충돌 동작 계획 알고리즘

$P_s(1)$ 에서  $P_g$ 까지 가기 위하여 n개의 슬라이스마다 2 절에서 언급한 문제를 고려하여  $FR(k)$  ( $k=1..n$ )을 회피하는 경로와 궤적을 구하는 알고리즘에 대해서 설명한다.

(단계 1)  $k \leftarrow 1$ ,  $t \leftarrow Vt$

(단계 2)  $P_s(k)$ 에서  $P_g$ 까지의 최단 경로를 V-그래프 탐색 알고리즘을 이용하여  $FR(k)$ 에 대해 구한다.

(단계 3) 경로를 따라 궤적 계획을 하여 k-슬라이스의 경유점  $P_{m'}(k)$ 를 구한다.

(단계 4) if ( $P_{m'}(k) \in FR(k+1)$ ), then goto 단계5  
else goto 단계7

(단계 5)  $FR(k+1)$ 에 대하여 새로운 경로를 구하고, 그 경로를 따라 궤적 계획을 하여 경유점  $P_{m'}(k+1)$ 을 구한다.

(단계 6) if ( $P_s(k) \in FR(k+1)$ ), and ( $P_{m'}(k+1) \in FR(k+1)$ )

then  $P_{m'}(k) \leftarrow P_{m''}(k+1)$  (그림 10(c))

else if ( $(P_s(k) P_{m'}(k+1)) \cap FR(k+1) \neq \emptyset$ , and ( $P_s(k) \notin FR(k+1)$ ))

then  $P_{m'}(k) \leftarrow P_{m''}(k+1)$  (그림 10(b))

else

$P_{m'}(k) \leftarrow P_{m'}(k+1)$  (그림 10(a))

(단계 7)  $t \leftarrow t + Vt$

if ( $P_{m'}(k) = P_g$ ) then goto 단계 8

if ( $t > T_r$ ) then  $P_s(k) \leftarrow P_{m'}(k)$  ( $T_r = n \times Vt$ )

else  $P_s(k+1) \leftarrow P_{m'}(k)$ ,  $k \leftarrow k+1$

goto 단계 2

(단계 8) stop

위의 상황뿐 아니라 몇가지 특수한 경우가 있는데, 각 경우와 대책은 다음과 같다.

(1) 로봇 R2의 동작 시점( $P_s(1)$ )이  $FR(1)$  내부에 있는 경우(그림 11(a)), 즉 직사각형으로 근사화하기 전의 FR인  $FR'(1)$ 의 외부에 있는 경우로서 실제로 충돌은 발생하지 않는 경우이다.

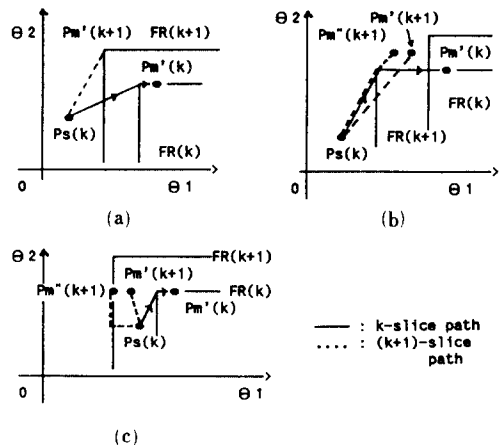


그림 10. k-슬라이스의 경유점이  $FR(k+1)$ 내에 있는 경우

Fig. 10. The via point in the k-th time slice lies in  $FR(k+1)$ .

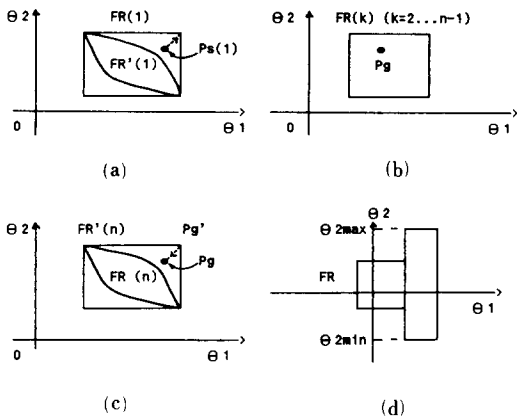


그림 11. 시점, 종점 및 링크 1의 충돌 발생 경우  
 Fig. 11. The case when the starting point, goal point and the 1-st link collide with FR.

첫 슬라이스에서 FR(1)의 모서리까지 이동하고, 이 점을 FR(2)와 비교한다.

(2) 로봇 R2의 종점이 슬라이스 중간에 있는 경우, 즉, Pg가 FR(k), k=2..n-1, 안에 있는 경우 (그림 11 (b)) :

k-슬라이스 까지 정지하고 이후에 동작을 한다.

(3) 로봇 R2의 종점이 FR(n)안에 있는 경우 (그림 11(c)) :

그림과 같이 가장 가까운 꼭지점이나 모서리의 점 (Pg')을 종점으로 하여 무 충돌 동작 계획 알고리즘을 수행하고, Pg'에서 Pg까지 간다.

(4) 로봇 R2의 링크 1이 충돌할 경우, 즉 k-슬라이스가 그림 11(d)와 같을 때 :

(2)와 같은 동작을 수행한다.

V. 검증 및 검토

제안된 알고리즘을 검증에 의해 확인함에 있어, view time을 변화시켜 그 영향을 살펴 본다.

(1) 검증 1

다음과 같은 검증 자료로 알고리즘의 수행 여부를 확인한다.

로봇 R1의 시점(S), 종점(G) : (500, -200), (450, 300) (mm)

로봇 R2의 시점(S), 종점(G) : (-60°, 45°), (50°, 5°)

T<sub>r</sub> = 4.0초, V<sub>t</sub> = 1.0초

● 검증 결과

V<sub>t</sub>가 1.0초이므로 네 개의 SR(영역 B 모델링),

FR 슬라이스가 구성된다(그림 12, 13). 그림 14는 알고리즘을 수행하지 않았을 때의 충돌 발생의 예를 보인 것이고, 그림 15는 FR에 대하여 알고리즘을 적용한 결과이다. 그림 (a)에서 1, 2, 3, 4는 근사시킨 FR을 동시에 도시한 것인데 번호 순대로 FR이 변화한다. 또한 a, b, c, d, e는 각 슬라이스마다 구한 경우점인데 이 점들을 직각 좌표계 상에 표시한 것이 그림 (b)의 1, 2, 3, 4, 5이다. 그림 (a)의 경로 P는 FR이 없는 상태에서의 로봇 R2의 경로인데 이를 직각 좌표계 상에 view time마다 표시한 것이 그림 14이다. 그림 (b)의 형상 2, 3, 4가 그림 14와 비교하여 회피 동작을 수행함을 알 수 있다. 여기서 형상 3, 4는 과도한 회피 동작을 하였는데 이는 FR을 직사각형으로 근사화함에 의해 발생된 결과로 볼 수 있다. 즉, 그림 15(a)의 FR 2, 3의 이상 모서리는 그림 13의 (b), (c)의 FR에 포함되어 있지 않은 영역이기 때문에 실제로 충돌 가능성이 없는 구역이기 때문이다. 한편 로봇 R1에 비해 로봇 R2의 같은 시간당 움직인 거리가 많은 것은 로봇 R2가 전 가속으로 운동하기 때문이다.

(2) 검증 2

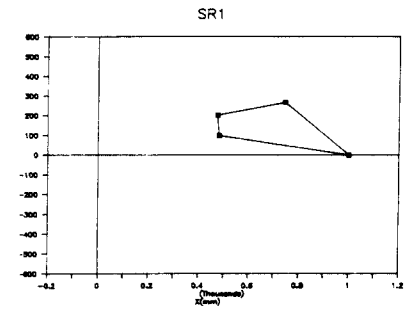
V<sub>t</sub>를 변화시켜 그 영향에 대해서 검토한다. 로봇 R1, R2의 시점과 종점은 검증 1과 같고, T<sub>r</sub>는 8.0초이다.

● 검증 결과

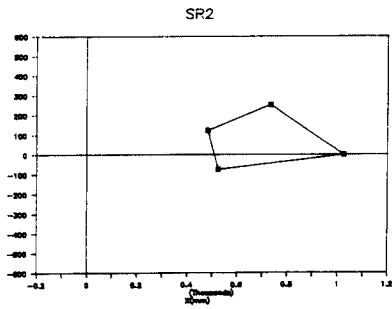
V<sub>t</sub>를 2.0초까지 변화시켜 검증을 수행한 결과, V<sub>t</sub>를 작게 할수록 원래의 경로로 근사해 가는 것을 알 수 있다(그림 16(a) - (d)). 특히, V<sub>t</sub>가 0.25초인 경우는 2.0초, 1.0초인 경우보다 과도한 회피 동작이 줄어들어 원래의 경로(그림 15(a)의 경로 P)에 접근되어 있음을 그림 17을 통해 알 수 있다. 이것은 V<sub>t</sub>가 감소할수록 FR의 제한 영역이 작아져서 그만큼 각 시간대에 로봇 R2가 움직일 수 있는 영역이 증가하기 때문이다.

지금까지 조인트 공간 상에서 충돌 가능성이 있는 두 로봇에 제안된 알고리즘을 적용한 결과 충돌 회피 동작을 수행함을 확인하였다. 그러나, 제안된 알고리즘은 효율적인 회피 동작을 수행하지 못하는 면이 있다. 이러한 문제를 해결하기 위하여 경로와 최적 계획을 할 때 전체 슬라이스를 고려하여 경우점을 구하는 방법을 찾아야 한다. 또한, FR을 근사화하는 과정에 있어 많은 조인트 제한이 있었기 때문에 과도한 회피 동작이 발생되었다. 이것은 FR을 근사화하는데 있어 원래의 FR 형태에 가까운 다각형으로 근사화하고, 일반적인 다각형 형태에 대한 무충

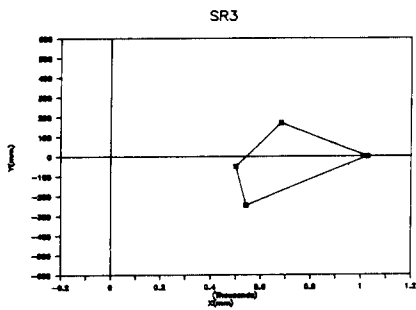




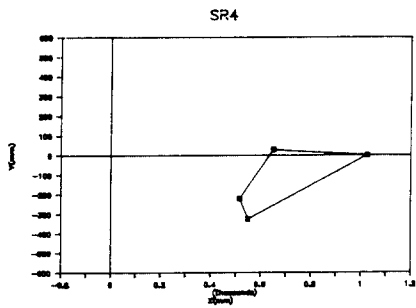
(a) 0.0-1.0 sec



(b) 1.0-2.0 sec

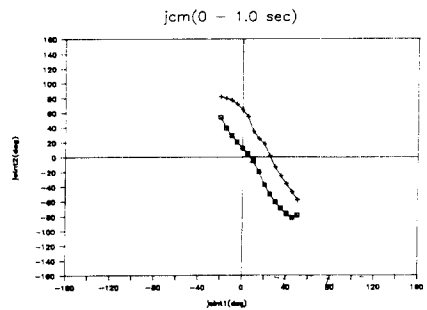


(c) 2.0-3.0 sec

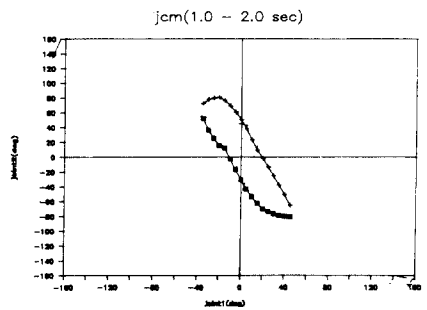


(d) 3.0-4.0 sec

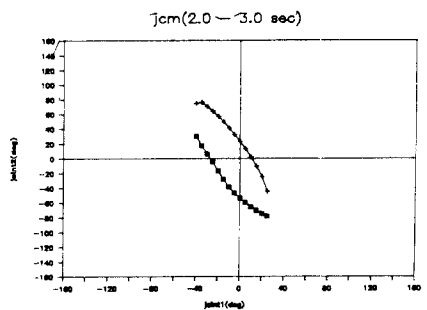
그림12. 구성된 SR(검증 1)  
Fig.12. The SR's (Simulation 1).



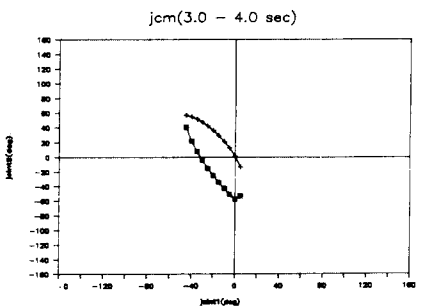
(a) 0.0-1.0 sec



(b) 1.0-2.0 sec



(c) 2.0-3.0 sec



(d) 3.0-4.0 sec

그림13. 구성된 FR(검증 1)  
Fig.13. The FR's (Simulation 1).

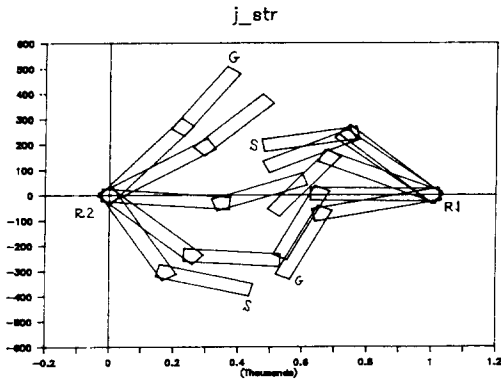
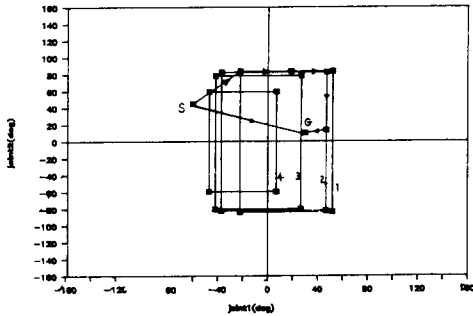
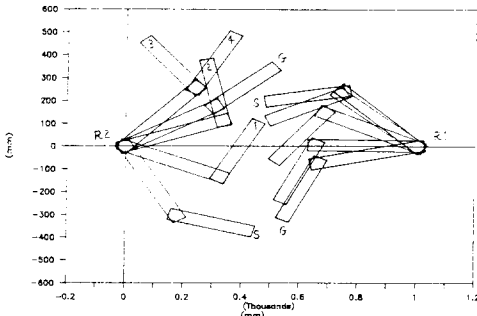


그림 14. 알고리즘 적용 전의 충돌 상황 (검증 1)  
Fig. 14. The collision with no collision-avoidance algorithm (Simulation 1).

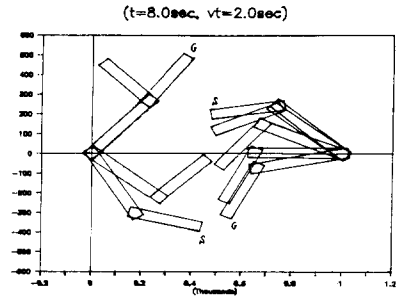


(a)

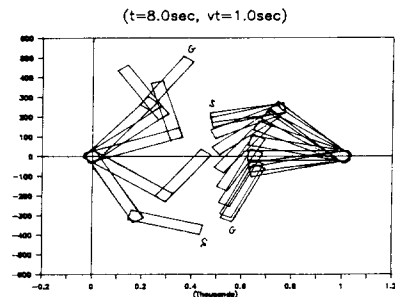


(b)

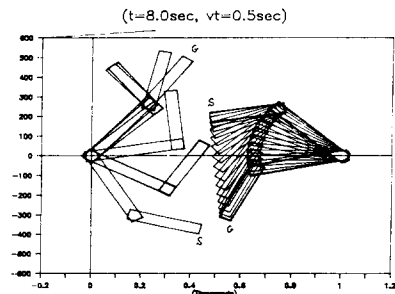
그림 15. 알고리즘 적용후의 회피 동작의 수행 (검증 1)  
(a) 조인트 좌표계 상의 회피 동작  
(b) 직각 좌표계 상의 회피 동작  
Fig. 15. The collision avoidance with the collision-avoidance algorithm (Simulation 1).  
(a) Collision avoidance in joint coordinate space.  
(b) Collision avoidance in Cartesian coordinate space.



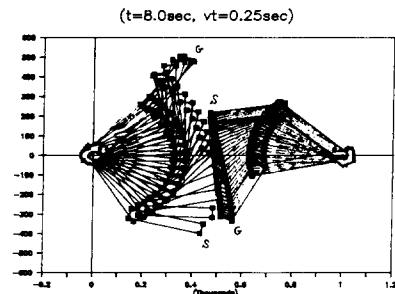
(a)  $V_t=2.0$  sec



(b)  $V_t=1.0$  sec



(c)  $V_t=0.5$  sec



(d)  $V_t=0.25$  sec

그림 16. view time 을 변경했을 때의 직각 좌표계 상의 궤적 변화 (검증 2)  
Fig. 16. The trajectory modified by the change of view time in the Cartesian coordinate space (Simulation 2).

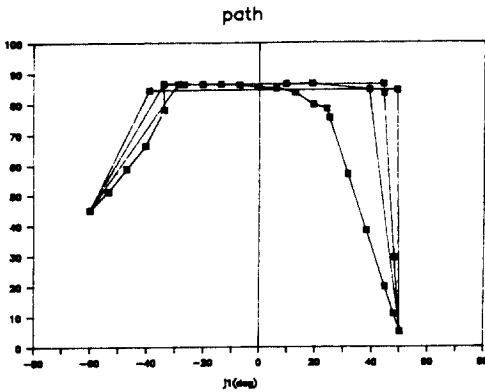


그림17. view time을 변경했을 때의 조인트 좌표계 상의 궤적 변화

Fig.17. The trajectory modified by the change of view time in the joint coordinate space.

둘 경로를 찾는 알고리즘으로 해결할 수 있다. 검증에 의해 확인된 바에 의하면  $V_t$ 를 줄임에 의해 이런 문제가 상당히 보완됨을 알 수 있었다. 한편, 각 슬라이스마다 구한 경유점을 통해서 로봇트가 움직이면 각 경유점에서 멈춰야 하는 문제점이 발생한다. 부드러운 동작(smooth motion)으로 각 경유점의 근방으로 충돌하지 않고 움직일 수 있는 방법이 요구된다.

### VI. 결 론

본 논문은 두 대의 2 자유도 평면 로봇트가 공동 작업 공간 내에서 작업을 할 때, 충돌 가능한 두 로봇트가 충돌 회피 동작을 하도록 하는 무충돌 동작 계획 알고리즘을 view time 개념을 적용하여 제안하였다. 본 논문은 세 개의 주요 부분으로 구성된다. 첫번째로 SR 구성 부분이고, 두번째는 JCM을 구성하는 부분이고, 세번째는 조인트 공간에서의 무충돌 동작을 계획하는 부분으로 구분할 수 있다.

SR 구성은, 직각 좌표계에서 직선 운동하는 로봇트의 궤적을 view time마다 이산화하여 각 시간마다 그 로봇트의 링크가 거처간 차위 영역을 다각형으로 모델링하여 다각형의 데이터 베이스를 구성하는 것이다. JCM 구성은, 매 viewtime마다 SR에 대한 로봇트 링크의 회전 접촉각을 구하여 접촉각의 분포를 조인트 좌표계 상에 표시한 지도를 만드는 것이다. 즉, 로봇트 링크간의 충돌을 고려하여 충돌 회피 동작을 할 수 있도록 하였다.

무충돌 동작 계획 부분은 다음과 같다. JCM 상의

FR이 조인트 좌표계상에서 불규칙 형태의 시변 물체이므로 FR을 직사각형 형태로 근사화하여 V-그래프 탐색 알고리즘을 적용할 수 있도록 하였으며, 시간 변화를 고려하여 회피 동작을 수행하는 알고리즘을 제안하였다. 그리고 이와같은 일련의 과정을 검증을 통해서 확인하였다.

앞으로 3 차원 물체에 대한 본 논문의 확대 연구와 세 대의 로봇트 시스템에 대한 적용이 필요하고, 다중 로봇트 시스템 설계, 다중 로봇트시스템의 효율적인 운용을 위하여 본 논문의 내용이 응용될 수 있으리라 판단된다.

### 參 考 文 獻

- [1] T. Lozano-Perez, "Spatial planning: a configuration space approach," IEEE Trans. on Computers. C-32(2):pp. 108-120, 1983.
- [2] R.A. Brooks and T. Lozano-Perez, "A subdivision algorithm in configuration space find-path with rotation," Report AIM-684, AI Lab, MIT, Cambridge, Massachusetts, Dec. 1982.
- [3] W.E. Red & H.V. Truong-Cao, "Configuration maps for robot path planning in two dimensions," ASME J. Dynamic Syst. Measur. Contr., vol. 107, Dec. 1985.
- [4] T. Lozano-perez and M.A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," Comm. ACM, vol. 10, no. 22, pp. 560-570, Oct. 1979.
- [5] K. Kant and S.W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," Int. J. Robotics Res., vol. 5, no. 3, pp. 72-89, Fall 1986.
- [6] J.W. Roach and M.N. Boaz, "Coordinating the motions of arms in a common workspace," IEEE J. Robotics and Automation, vol. RA-3, no. 5, pp. 437-444, Oct. 1987.
- [7] J.T. Schwartz and M. Sharir, "On the piano movers' problem:III coordinating the motion of several independent bodies: The special case of circular bodies amidst polygonal barriers," Int. J. of Robotics Reserch, 2(3), pp. 46-75, 1983.
- [8] M. Erdmann and T. Lozano-Perez, "On multiple moving objects," in Proc. IEEE Int. Conf. Robotics and Automation, pp. 1419-1424, 1986.

- [9] W.T. Park, "State-space representations for coordination of multiple manipulators," in Proc. 14th Int Symp. Industrial Robots, Gothenburg, Sweden, pp. 397-405, 1984.
- [10] B.H. Lee, "Wrist collision avoidance of two robots: a collision map and time scheduling approach," in Proc. 25th IEEE Conf. Decision and Control, Athens, Greece, pp. 429-434, Dec. 1986.
- [11] B.H. Lee, "An approach to motion planning and motion control of two robots in a common workspace," Ph. D Dissertation, pp. 26-28, 1985.
- [12] 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, Sept. 1983.
- [13] T. Lozano-perez, "A simple motion-planning algorithm for general robot manipulators," IEEE J. of Robotics and Automation, vol. RA-3, no. 3, pp. 224-237, June 1987.
- [14] 남윤석, 공동 작업 공간을 갖는 두 로봇의 충돌 회피 동작에 관한 연구, 서울대학교 제어계측공학과, 석사 학위 논문, pp. 24-29, 1989.

著 者 紹 介



**南 潤 錫 (正會員)**  
 1963年 9月 11日生. 1987年 2月 서울대학교 제어계측 공학과 졸업(공학사). 1989年 2月 서울대학교 대학원 제어계측 공학과 졸업(공학석사). 1989年 3月 ~ 현재 서울대학교 대학원 제어계측공학과 박사과정. 주관심분야는 로보틱스 및 자동화 등임.



**高 明 三 (正會員)**  
 1930年 1月 1日生. 1955年 서울대학교 전기공학과 졸업(공학사). 1967年~1969年 미국 뉴 멕시코 대 대학원. 1972年 서울대학교 대학원 전기공학과 졸업(공학박사) 1962年~1978年 서울대학교 공과대학 전기공학과 교수. 1979年~현재 서울대학교 공과대학 제어계측공학과 교수. 주관심분야는 자동제어, 로보틱스 등임.

**李 範 熙 (正會員)** 第 26 卷 第 3 號 參照  
 현재 서울대학교 제어계측공학과 조교수



**高 樂 溶 (正會員)**  
 1961年 3月 13日生. 1985年 2月 서울대학교 제어계측 공학과 졸업(공학사). 1987年 2月 서울대학교 대학원 제어계측공학과 졸업(공학석사). 1989年 8月 서울대학교 대학원 제어계측 공학과 박사과정 수료. 주관심분야는 로보틱스 및 자동화 등임.