

# 확장충돌맵의 수학적 분석을 이용한 다개체의 충돌탐지

## Conflict Detection for Multi-agent Motion Planning using Mathematical Analysis of Extended Collision Map

윤영환<sup>1</sup> · 최정식<sup>2</sup> · 이범희<sup>3</sup>

Y.H.Yoon<sup>1</sup> · J.S.Choi<sup>2</sup> · B.H.Lee<sup>3</sup>

**Abstract** Effective tools which can alleviate the complexity and computational load problem in collision-free motion planning for multi-agent system have steadily been demanded in robotics field. To reduce the complexity, the extended collision map (ECM) which adopts decoupled approach and prioritization is already proposed. In ECM, the collision regions which represent the potential collision of robots are calculated using the computational power; the complexity problem is not resolved completely. In this paper, we propose a mathematical analysis of the extended collision map; as a result, we formulate the collision region as an equation with 5–8 variables. For mathematical analysis, we introduce realistic assumptions as follows; the path of each robot can be approximated to a straight line or an arc and every robot moves with uniform velocity or constant acceleration near the intersection between paths. Our result reduces the computational complexity in comparison with the previous result without losing optimality, because we use simple but exact equations of the collision regions. This result can be widely applicable to coordinated multi-agent motion planning.

**Keywords** : extended collision map, multi-agents, conflict detection, collision-free control

### 1. 서 론

로봇의 동작 계획은 공간상에서의 이동 경로의 계획과 시간에 따른 궤적의 계획으로 구성된다<sup>1</sup>. 무충돌 동작 계획을 위해서는 로봇간의 잠재적인 충돌을 탐지하고 이를 피할 수 있도록 제어해야 하는데, 이를 충돌문제의 해결(conflict resolution)이라고 한다. 충돌문제의 해결은 다개체 로봇의 동작계획에 있어 핵심적인 기술 중 하나라 할 수 있으며, 기존의 다양한 연구에서는 아래와 같은 여러 접근 방향에서 이 문제를 해결할 수 있는 기법을 제하고 있다.

로봇간 충돌을 피할 수 있는 이동 경로와 궤적을 다개체에 대해 동시에 해결하는 것(centralized approach)은 매우 어려운 문제이다<sup>1</sup>. 따라서 상당수의 연구에서는 분리식 접근(decoupled approach)을 채택하고 있는데, 이는

동작 계획을 (공간상에서의) 이동 경로의 계획과 (시간에 따른) 궤적의 계획으로 분리하여 각각 해결하는 접근 방식이다<sup>2, 6, 8</sup>. 다른 방식으로 각 개체가 독립적으로 판단하고 이동하는 분산식 접근(distributed approach)이 제안되었다<sup>5, 10</sup>. 위와 같은 전통적인 접근 방식 이외에도, 시장 기반(market-based) 기법<sup>7</sup> 및 각종 혼합 기법들이 제안되었다.

로봇 간 무충돌 해를 구하기 위한 몇몇 구체적인 연구로부터 실용적인 도구들이 개발된 바 있다. [3]에서는 충돌맵(collision map) 기법을 제안했는데, 이는 동일한 작업 공간 내에 있는 두 대의 로봇 사이에서 잠재적인 충돌을 효과적으로 탐지할 수 있는 도구이다. 본 논문에서는 충돌맵을 기본적인 충돌문제 해결 도구로 채택하였다. 충돌맵 기법의 장점은 잠재적인 충돌에 대한 정보를 2차원 그래프를 통해 시각적으로 보여준다는 점이다. 이 2차원 그래프는 작업공간 내에서의 잠재적인 충돌을 표시하는 충돌영역을 포함한다. [9]에서는 최근 연구에서 기하학적 분석을 통해 충돌맵의 특성을 자세히 보여준 바 있다. 충돌맵을

<sup>1</sup> 서울대학교 전기컴퓨터공학부 석사과정  
(E-mail: hwan099@snu.ac.kr)

<sup>2</sup> 서울대학교 전기컴퓨터공학부 박사과정 (E-mail: jsforce@snu.ac.kr)

<sup>3</sup> 서울대학교 전기컴퓨터공학부 교수 (E-mail: bhlee@snu.ac.kr)

이용하여 충돌 문제를 해결하는 기법 또한 기존 연구에서 제안되었다<sup>[2]</sup>. 최적의 충돌회피를 위한 궤적을 구하기 위해 [4]에서는 최소시간지연(minimum time delay) 기법의 개념을 제안하였다.

충돌맵 기법은 원래 두 대의 로봇 사이에서 충돌 문제를 해결하기 위해 개발한 도구이다. 두 대의 로봇이라는 한계를 극복하고 다개체 상황에서의 충돌 문제를 해결하기 위해, 확장충돌맵(extended collision map) 기법이 제시되었다<sup>[11]</sup>. 이 연구에서는 확장충돌맵 기법에서 최소시간지연을 적용하여 충돌문제를 해결하는 방법을 제시하였는데, 이는 컴퓨터의 계산력을 이용한 것이다.

그러나 하나의 통제시스템에서 다개체 상황에서의 충돌 문제를 해결하는 방식에서는 해를 구하기 위해 많은 계산량을 필요로 한다. 계산량 문제는 확장충돌맵과 같은 도구에서 잠재적인 충돌을 수학적으로 표현하지 못한 것에 크게 기인하고 있다. 따라서 본 논문에서는 확장충돌맵에서 나타나는 잠재적인 충돌을 수학적으로 분석하고 표현하여, 앞으로 이를 통해 도출한 수식을 이용하여 충돌문제를 좀 더 빠르게 해결할 수 있도록 하고자 한다.

2장에서는 충돌맵 및 확장충돌맵 기법을 간략하게 소개한다. 3장에서는 확장충돌맵 내의 충돌영역을 수학적으로 분석하고 충돌 탐지에 대해 살펴본다. 이어 4장에서는 수학적 분석을 적용한 다개체 충돌탐지 모의실험 결과를 간략하게 보인다. 마지막으로 5장에서는 결론을 도출한다.

## 2. 충돌맵 및 확장충돌맵 기법

### 2.1 충돌맵(collision map)<sup>[3]</sup>

충돌맵은 경로가 결정된 후 발생가능한 충돌을 검사하는 분리식 방식을 채택하였다. 경로가 결정되어 있기 때문에, 로봇의 위치 정보는 경로상의 1 차원 정보로 표시될 수 있다. 따라서 로봇의 시간에 따른 경로상의 위치(이동거리)를 시간-거리 그래프 상에 표시할 수 있다.

단순화를 위해, 그림 1 에서처럼 두 로봇의 반지름을 더한 가상원(virtual circle)을 설정한다. 가상원의 중심은 선순위 로봇의 중심과 같다. 그러면 선순위 로봇을 가상원으로, 후순위 로봇을 한 점으로 간주할 수 있다.

그림 1 에서처럼 두 로봇의 경로가 교차하게 되면 잠재적인 충돌 위험이 발생하게 된다. 선순위 로봇이 후 순위 로봇의 경로를 지나가는 것을 시간에 따라서

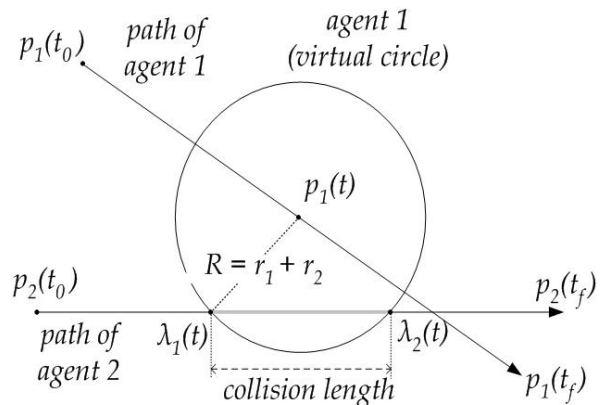


그림 1. 가상원 및 잠재적인 충돌 위험

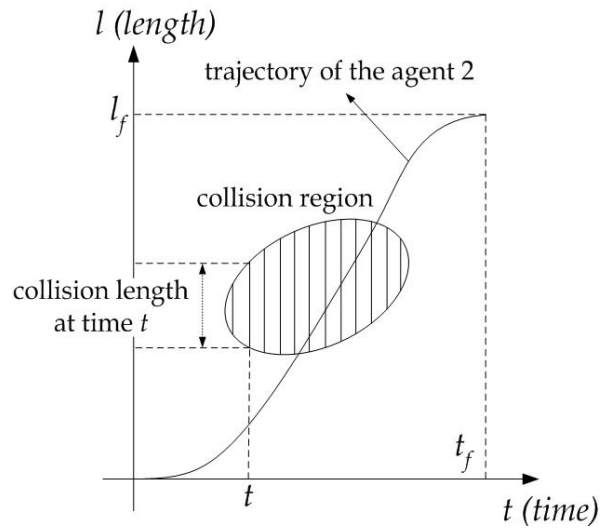


그림 2. 충돌맵 내의 충돌영역

나타내면 그림 2 에서와 같은 영역이 나타나는데, 이를 충돌영역(collision region)이라고 한다. 로봇의 궤적이 충돌영역을 통과하면 두 로봇은 궤적이 충돌영역에 들어가는 시간에 충돌하게 된다.

로봇의 궤적이 충돌영역을 지나가지 않도록 하는 충돌회피 기법으로 출발시간을 늦추는 시간지연, 속도를 줄이는 감속, 그리고 시간지연 방법을 수정한 최소시간지연 등이 있다(그림 3).

### 2.2 확장충돌맵(extended collision map) [11]

확장충돌맵은 앞의 충돌맵을 다개체의 경우로 확장한 도구로 볼 수 있다. 확장충돌맵에서는 로봇의 우선순위가 결정되어 있으며, 각 로봇은 자신보다 우선순위가 높은 모든 로봇들과 생기는 충돌영역들을 모두 포함하는 확장충돌맵을 가진다.

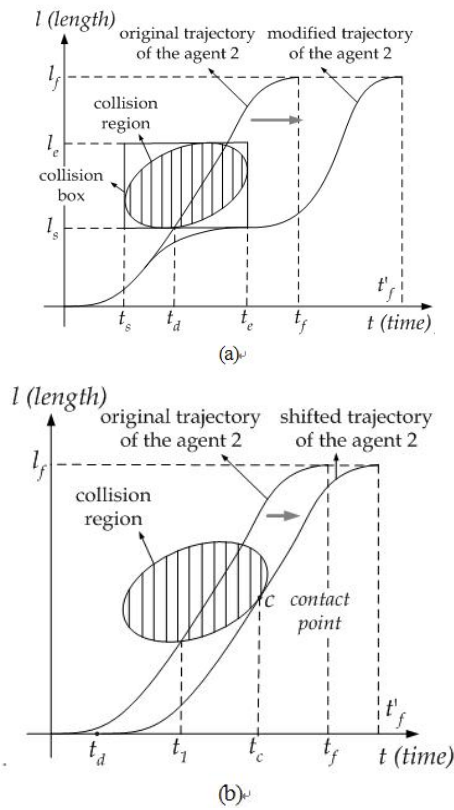


그림 3. 충돌회피기법: (a) 감속; (b) 최소시간지연

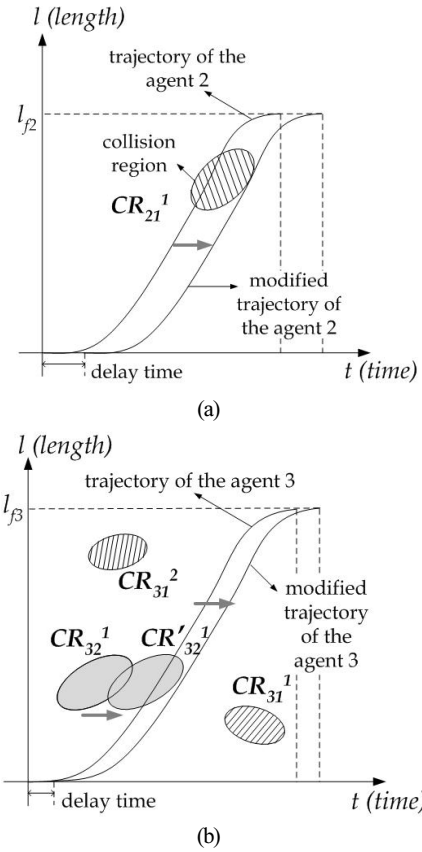


그림 6. 확장충돌맵을 이용한 충돌회피: (a) 로봇 2의 수정된 확장충돌맵; (b) 로봇 3의 수정된 확장충돌맵

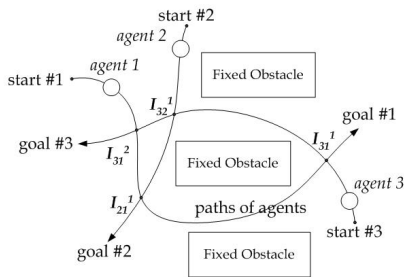


그림 4. 동일 작업공간 내 3대 로봇의 주행

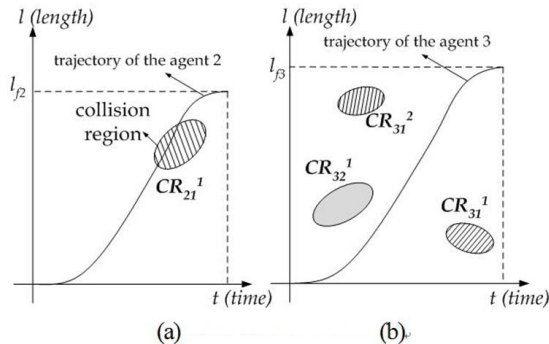


그림 5. 확장충돌맵: (a) 로봇 2의 확장충돌맵; (b) 로봇 3의 확장충돌맵 (CR = Collision Region)

그림 4 와 같이 세 대의 로봇이 주행하는 경우, 우선순위가 낮은 로봇 2와 로봇 3의 확장충돌맵은 그림 5 과 같다.

그림 5 의 왼쪽 그림은 로봇 1 과 로봇 2 가 충돌할 것임을 나타내고 있다. 확장충돌맵을 이용한 충돌해결은 우선순위를 부여하여 순차적으로 해결하는 방식을 사용하였다. 앞의 예와 같은 경우, 로봇 2 의 궤적을 바꾸게 되면 로봇 3 의 확장충돌맵에서 로봇 2 에 의해 발생하는 충돌영역이 변하게 된다. 변화된 확장충돌맵을 이용하여 로봇 3 의 충돌을 탐지하고, 충돌이 예상되는 경우 궤적을 수정할 수 있다. 이와 같은 방식을 적용하여 충돌문제를 해결한 결과는 그림 6 과 같다.

### 3. 충돌영역의 수학적 분석 및 충돌탐지

본 논문에서는 각 로봇의 경로가 직선 또는 원호(arc) 형태이거나, 또는 그러한 형태로 근사화될 수 있다고 가정한다. 이러한 가정은 전 경로에 대해 성립할 필요는 없으며, 충돌이 발생할 수 있는 (공간상에서 둘 이상의

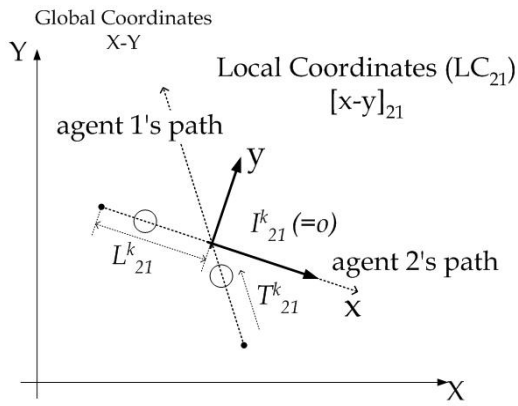


그림 7. 로컬 좌표계의 설정

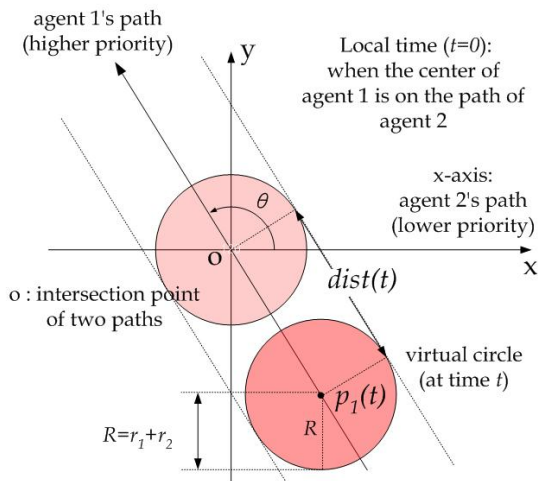


그림 8. 두 직선 경로의 교차

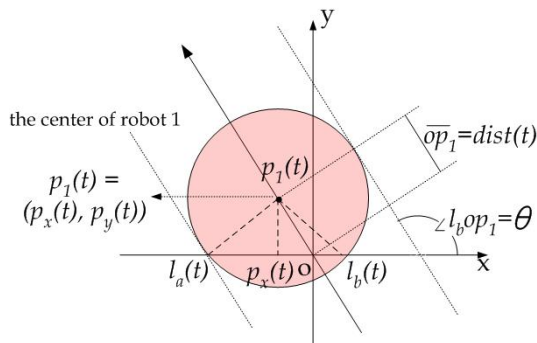


그림 9. 시간 t에서의 가상원 및 충돌분석

경로가 교차하는) 교차영역에서만 적용 가능하면 된다. 수식적 간결성을 위해, 각 로봇은 충돌지역 근처에서 등속운동 또는 등가속운동을 하는 것으로 가정한다. 이 조건 또한 교차영역에서만 적용 가능하면 된다.

이러한 가정을 바탕으로, 두 로봇 사이의 충돌영역을

수학적으로 분석하여 수식적 결과를 제시하고자 한다. 먼저 두 경로가 모두 직선인 경우, 한 경로가 원호 형태를 가지는 경우, 두 경로가 원호 형태를 가지는 경우 충돌영역의 수식을 제시하기로 한다. 본 논문에서도 이전 연구와 같이 우선순위의 의한 운동계획 방법을 따르고 있다.

### 3.1 두 직선 경로가 교차하는 경우

먼저, 수학적 표현을 위해 두 경로의 교점을 중심으로 하는 로컬 좌표계를 설정한다. 그림 7 에서와 같이, 로봇 2 의 경로 방향을 로컬 좌표계의 x 축으로 설정한다. 또한 로봇 1 이 교점(로컬 좌표계의 원점)에 위치할 때의 시간을 로컬시간  $t = 0$  으로 한다. 따라서 여기서의 시간은 로봇 1 이 교점을 통과하기 전에는 음수값을 가지게 되며, 교점을 통과한 후에는 양수값을 가진다.

그림 8 에서 거리함수  $dist(t)$ 는 로봇 1 과 로봇 2 의 반지름을 합산한 가상원 중심과 원점 사이의 거리로 정의한다. 예를 들어 (로봇 1 이) 등속도  $v$  로 움직이는 경우, 거리함수  $dist(t)$ 는  $dist(t) = |vt|$ 로 표현된다. 등가속도  $a$  로 움직이는 경우에는  $dist(t) = |vt + 0.5at^2|$ 이다. 여기서  $v$  의 값은 로봇 1 이 교점 위에 있을 때의 속도이다.  $\theta$  는 로봇 2 의 경로 방향인 x 축 의 양의 방향에서 로봇 1 의 경로 방향에 이르는 각으로 정의한다.

그림 9 에서 가상원과 로컬좌표계의 x 축 사이의 교점 좌표  $l_a(t), l_b(t)$ 는 다음과 같다.

$$l_a(t) = p_x(t) - \sqrt{R^2 - p_y^2(t)}, \quad (1)$$

$$l_b(t) = p_x(t) + \sqrt{R^2 - p_y^2(t)}$$

로봇 1 의 중심  $p_1(t)$ 의 좌표  $p_x(t), p_y(t)$ 는 식 (2)와 같다.

$$p_x(t) = dist(t) \cdot \cos \theta \cdot \text{sgn}(t)$$

$$p_y(t) = dist(t) \cdot \sin \theta \cdot \text{sgn}(t), \quad \text{sgn}(t) = \begin{cases} 1 & \text{where } t \geq 0 \\ -1 & \text{elsewhere} \end{cases} \quad (2)$$

그림 9 에서 로봇 2 가 자신의 경로상의  $l_a(t)$ 와  $l_b(t)$  사이에 위치하면 두 로봇 사이에 충돌이 발생하게 된다. 따라서 시간 t 에서의 충돌영역은  $l_a(t) \leq l \leq l_b(t)$ 와 같다. 시간 t 에 따라 교점좌표를 누적시키면 전체 충돌영역을 그림 10 과 같이 구할 수 있고 이 영역은 수식으로 표현되게 된다.

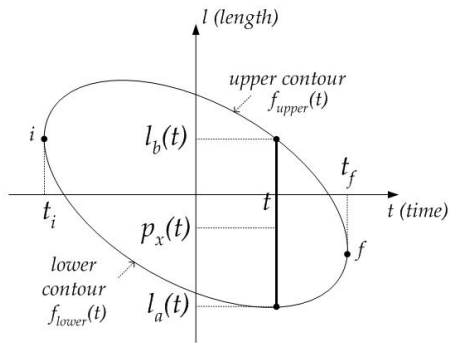


그림 10. 충돌영역 (로컬좌표계에서)

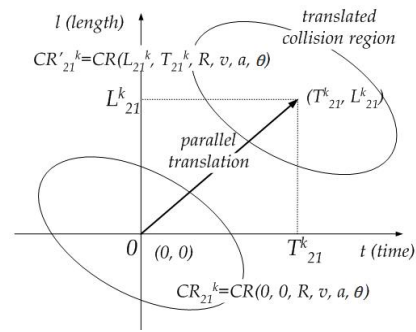


그림 11. 충돌영역의 평행이동

충돌영역은 두 교점 좌표의 연속형태로 표현되므로 위쪽 경계를  $f_{upper}$ , 아래쪽 경계를  $f_{lower}$  라고 하자.

$$\begin{aligned} f_{lower}(t) &= l_a(t), f_{upper}(t) = l_b(t) \\ CR &= \{(t, l) \mid f_{lower}(t) \leq l \leq f_{upper}(t)\} \end{aligned} \quad (3)$$

지금까지 충돌영역 경계는 로컬 좌표계에서 이루어진 것이다. 따라서 이 충돌영역을 전체 관점(global coordination)에서 표현해야 한다. 그림 7에서 ( $t = 0$  으로 두었던) 로봇 1 이 교점 위에 위치하는 시점은 전체 좌표계에서는 로봇이 이동을 시작한 후  $T_{21}^k$  만큼 시간이 지난 후이며, 로컬 좌표계의 원점으로  $l = 0$  으로 잡았던 위치는 로봇 2 의 이동거리  $L_{21}^k$  에 해당하는 곳임을 알 수 있다. 따라서 위의 그림 10 에서와 같은 충돌영역을 그림 11 와 같이 평행이동해야 전체 좌표계에서 충돌영역을 표현할 수 있다.

이상의 논의에서, 충돌영역은 등속운동하는 경우에 모두 5 개의 변수로 표현됨을 알 수 있다. 평행이동 전에 식 (1), (2)에서  $v, R, \theta$  의 세 변수가 필요하며, 평행이동에  $L_{21}^k, T_{21}^k$  의 두 변수가 필요하다. 이 중  $L_{21}^k, T_{21}^k, \theta$  는 공간상에서 로봇의 경로에 따라 결정되며,  $v$  와  $R$  은 로봇의 물리적 특성에 의해 결정된다. 5 개의 변수로 충돌영역을 표시할 수 있으므로, 충돌영역을 아래와 같이 함수 형태로 표현할 수 있다.

$$CR_{21}^k = CR_1(L_{21}^k, T_{21}^k, v, R, \theta) \quad (4)$$

$$\begin{aligned} CR_{21}^k &= \{(l, t) \mid L_{21}^k + f_{lower}(t - T_{21}^k) \leq l \leq \\ &L_{21}^k + f_{upper}(t - T_{21}^k), T_{21}^k + t_i \leq t \leq T_{21}^k + t_f\} \end{aligned} \quad (5)$$

식 (4)에서 CR 의 첨자 ‘1’은 충돌조건이 다른 경우

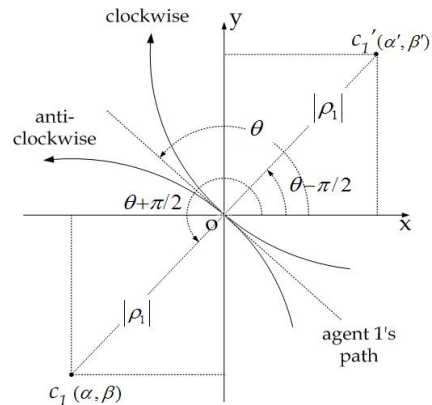


그림 12. 로봇 1의 경로가 원호인 경우

필요한 변수가 달라지는 것을 구별하기 위한 것이다.

만일, 로봇 1 이 등가속도 운동을 하는 경우에는 가속도  $a$  를 포함하여 식 (6)에서와 같이 6 개의 변수가 필요하게 된다.

$$CR_{21}^k = CR_{1A}(L_{21}^k, T_{21}^k, v, a, R, \theta) \quad (6)$$

### 3.2 한 경로가 원호인 경우

로봇이 원호의 경로를 따라 이동하는 경우에도 유사한 방법을 사용한다. 두 로봇의 교점을 로컬 좌표계의 원점으로, 교점에서 로봇 2 의 이동 방향(경로의 접선방향)을 x 축의 양의 방향으로 설정한다. 로봇 1 이 곡선을 따라 이동하는 경우에는 거리함수  $dist(t)$  에 대응되는 개념으로, 교점을 기준으로 하여 경로를 따라 회전한 회전각함수  $\varphi(t)$  를 사용한다.

로봇이 원호의 경로를 따라 이동하는 경우에는 원호의 곡률반경을 표현하기 위한 추가 변수가 필요하게 된다. 로봇 1 과 로봇 2 의 경로에 대한 곡률반경을 각각  $|\rho_1|, |\rho_2|$  이라고 한다. 동일한 곡률반경을 가지는 경로가 그림 12 에서와 같이 공간상에서 두 개씩 존재하므로, 로봇

$i$ 의 경로가 반시계 방향과 시계 방향으로 회전할 때 각각  $\rho_i > 0, \rho_i < 0$ 로 하여 두 경우를 구별하기로 한다.

로봇 1의 경로가 원호이고 로봇 2의 경로가 직선인 경우 평행이동 전의 충돌영역의 경계가 가지는 식은 아래와 같다.

$$\begin{aligned} l_a(t) &= p_x(t) - \sqrt{R^2 - p_y^2(t)}, \\ l_b(t) &= p_x(t) + \sqrt{R^2 - p_y^2(t)} \end{aligned} \quad (7)$$

$$\begin{bmatrix} p_x(t) \\ p_y(t) \end{bmatrix} = \begin{bmatrix} 1 - \cos \varphi(t) & \text{sgn}(\rho_1) \cdot \sin \varphi(t) \\ -\text{sgn}(\rho_1) \cdot \sin \varphi(t) & 1 - \cos \varphi(t) \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (8)$$

$\alpha = \rho_1 \sin \theta, \beta = -\rho_1 \cos \theta, \varphi(t) = \omega t$

위 식은 로봇 1이 원호 경로의 중심을 기준으로 각속도 상수  $\omega$ 로 돌면서 이동하는 경우이다. 이 경우에는 6개의 변수로 아래의 식(9)과 같이 표현할 수 있다. 등가속도 운동을 하는 경우 각가속도를 포함한 7개의 변수가 필요하다.

$$CR_{21}^k = CR_2(L_{21}^k, T_{21}^k, \omega, R, \theta, \rho_1) \quad (9)$$

### 3.3 두 경로가 원호인 경우

두 경로 모두 원호인 경우에 로컬좌표계에서 충돌영역의 경계가 가지는 식은 아래와 같다.

$$\begin{aligned} f_{lower}(t) &= -|\rho_2|(\phi/2 + \varepsilon), \\ f_{upper}(t) &= +|\rho_2|(\phi/2 - \varepsilon) \end{aligned} \quad (10)$$

$$\phi = 2 \arccos \frac{2(p_y - \rho_2) \cdot \rho_2 - (p_x^2 + p_y^2 - R^2)}{2|\rho_2| \sqrt{p_x^2 + (p_y - \rho_2)^2}} \quad (11)$$

$$\varepsilon = \arctan \frac{p_x \cdot \text{sgn}(\rho_2)}{p_y - \rho_2} \quad (12)$$

$$\begin{bmatrix} p_x(t) \\ p_y(t) \end{bmatrix} = \begin{bmatrix} 1 - \cos \varphi(t) & \text{sgn}(\rho_1) \cdot \sin \varphi(t) \\ -\text{sgn}(\rho_1) \cdot \sin \varphi(t) & 1 - \cos \varphi(t) \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (13)$$

$\alpha = \rho_1 \sin \theta, \beta = -\rho_1 \cos \theta, \varphi(t) = \omega t$

이 식은 로봇 1이 앞에서의 경우와 마찬가지로 등속력 운동을 할 경우이며, 식(14)와 같이 총 7개의 변수가 필요하다. 등가속력인 경우 8개가 필요하다.

$$CR_{21}^k = CR_3(L_{21}^k, T_{21}^k, \omega, R, \theta, \rho_1, \rho_2) \quad (14)$$

### 3.4 충돌탐지 및 충돌영역의 예

다음 순서에 따라 각 로봇의 확장충돌맵을 작성하여 잠재적인 충돌을 탐지할 수 있다.

- (1) 교차영역을 직선/원호 경로로 근사
- (2) 변수값 결정
- (3) 충돌영역 경계식 도출
- (4) 충돌영역과 로봇 궤적으로 잠재적 충돌을 탐지함

특히 가속-등속-감속과 같이 로봇 궤적이 수직화되는 경우에는 궤적과 충돌영역의 교차여부를 수학적으로 직접 구할 수 있다.

경로가 정해진 두 로봇으로부터 충돌조건을 도출하여 충돌영역을 나타내는 예를 그림 13과 그림 14에 제시하였다. 각 충돌영역은 앞에서 제시한 수식(1-2), (7-8)을 이용하여 구한 결과이다.

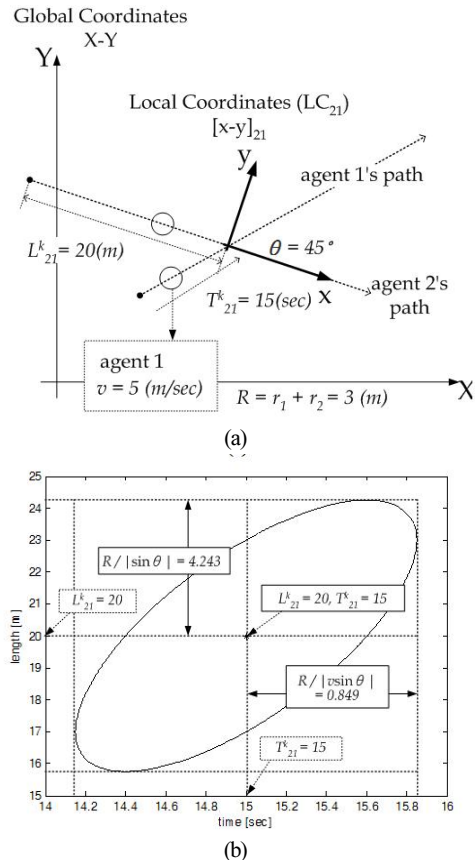
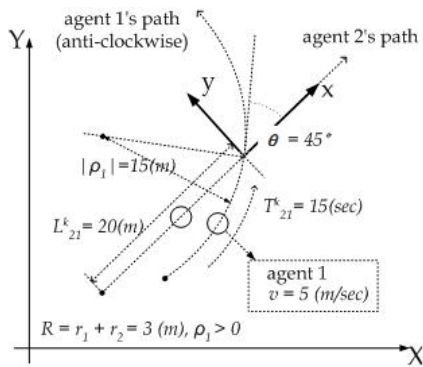
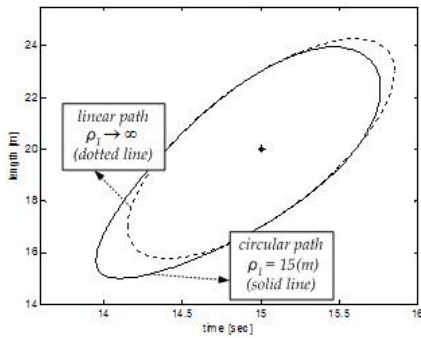


그림 13. 두 직선경로의 교차 예: (a) 공간상에서의 경로 교차; (b) 그에 따른 충돌영역



(a)



(b)

그림 14. 직선경로와 원호경로의 교차 예: (a) 공간상에서의 경로 교차; (b) 그에 따른 충돌영역

#### 4. 모의 실험

그림 15 에서와 같이, 12 대의 로봇이 동일 작업 공간 내에서 이동할 때의 모의 실험을 수행하였다. 앞의 수학적 분석 결과를 이용하여 각 로봇의 확장충돌맵을 만들고, 충돌회피 기법으로 최소시간지연 방식을 사용하였다.

그림 16 은 다섯 대의 로봇에 대해 충돌을 탐지한 결과이다. 수식을 이용하여 우측의 확장충돌맵에 충돌영역을 표시하였다. 로봇의 궤적이 충돌영역을 통과하므로 잠재적인 충돌이 탐지된다.

그림 17 은 최소시간지연을 이용하여 충돌을 회피한 결과이다. 최적해를 도출하여, 각 로봇들이 스치듯 충돌을 회피할 수 있었다. 이처럼 수학적 분석을 이용하여 충돌영역을 표현하고, 이를 충돌회피 기법에 활용하여 무충돌 주행 제어를 할 수 있다.

#### 5. 결 론

공간상의 교차영역 부근에서 경로를 직선 또는 원호

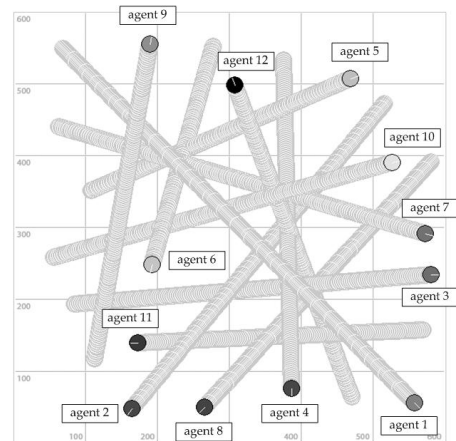


그림 15. 모의실험: 12대의 로봇에 대한 충돌탐지

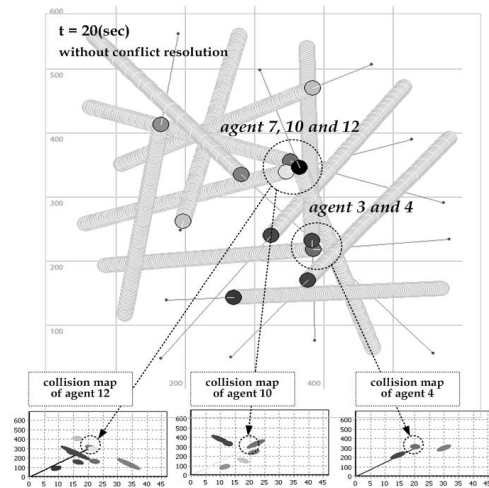


그림 16. 충돌회피 전: 로봇들 간의 충돌탐지

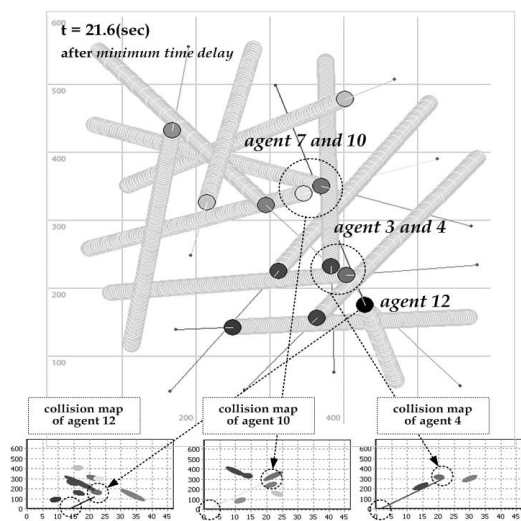


그림 17. 충돌회피 후: 최소시간지연 적용

형태로 근사화하고 로봇이 동일한 속력 또는 동일한 가속력으로 이동한다는 것을 가정하여 충돌영역의 수식을 제시할 수 있었다. 이 결과를 이용하여 다개체 로봇간의 잠재적인 충돌을 탐지할 수 있다.

기존에 확장충돌맵 내의 충돌영역을 표현하고 다루던 방식에서는 다음과 같은 문제가 있었다: 충돌상자로 단순화하는 경우에는 최적해를 찾지 못하였으며, 계산력을 이용하는 경우에는 많은 계산량을 필요로 하였다. 충돌영역을 수식으로 표현하게 되면 적은 계산량으로 빠르게 결과를 얻을 수 있으며, 동시에 계산력을 이용하는 경우에서처럼 최적해를 얻을 수 있는 장점이 있다.

확장충돌맵의 수학적 분석을 이용한 충돌탐지 기법은 빠른 충돌문제 해결 기법에 적용될 수 있다. 이는 실시간 제어가 필요한 경우나 충돌조건이 바뀌는 경우 등, 다개체를 이용하는 여러 상황을 빠르게 해결할 수 있는 도구가 될 것이다.

참고문헌

[1] J.E.Hopcroft, J.T.Schwartz, and M.Sharir, "On the Complexity of Motion Planning for Multiple Independent Object: PSPACE-Hardness of the Warehouseman's problem," International Journal of Robotics Research, 3(4):76-88, 1984.

[2] K. Kant and S.W.Zucker, "Toward Efficient Trajectory Planning: The Path-Velocity Decomposition," The International Journal of Robotic Research, Vol.5, No.3, pp.72-89, Fall 1986.

[3] B.H.Lee, C.S.G.Lee, "Collision-free Motion Planning of Two Robots," IEEE Transactions on Systems, Man, and Cybernetics, vol. 17, no 1, pp. 21-31, January/February, 1987.

[4] C.Chang, M.J.Chung, and B.H.Lee, "Collision Avoidance of Two General Robot Manipulators by Minimum Delay Time," IEEE Transactions on Systems, Man, and Cybernetics, vol. 24, no. 3, pp. 517-522, March, 1994.

[5] K.Azarm and G.Schmit, "Conflict-free Motion of Multiple Mobile Robots Based on Decentralized Motion Planning and Negotiation," Proc. of IEEE Int. Conf. on Robotics and Automation, April, 1997.

[6] C.Belta, and V.Kumar, "Motion Generation for Formations of Robots: A Geometric Approach," IEEE of Conf. Robot and Automation, Seoul, Korea, 2001.

[7] M.B.Dias, and A.Stentz, "A Market Approach to Multirobot Coordination," Technical report, CMU-RI-TR-01-26, Robotics Institute, Carnegie Mellon Univ. August, 2001.

[8] M.M.Quottrup, T.Bak, and R.I.Zamanabadi, "Multi-robot Planning: A Timed Automata Approach," Proc. of IEEE Int.

Conf. on Robotics and Automation, New Orleans, LA, 2004.

[9] S. H. Park and B. H. Lee, "Analysis of Robot Collision Characteristics Using the Concept of the Collision Map," Robotica, Vol. 24, pp. 295-303, May 2006.

[10] D.L.Wang, D.K.Liu, and G.Dissanayake, "A variable speed force field method for multi-robot collaboration," Proc. of IEEE Int. Conf. on Intelligent Robots and Systems, pp. 2697-2702, Beijing, China, 2006.

[11] S.H.Ji, J.S.Choi and B.H.Lee, "A Computational Interactive Approach to Multi-agent Motion Planning," International Journal of Control, Automation, and Systems, vol.5, no.3, pp.295-306, June, 2007.



윤영환

2005 서울대학교 전기공학부 (학사)  
 2005~현재 서울대학교 전기 컴퓨터공학부 석사과정  
 관심분야: Motion planning, Multi-agent system



최정식

1998 서울대학교 농공학과 기계전공 (학사)  
 2004 서울대학교 전기컴퓨터공학부 (석사)  
 2004~현재 서울대학교 전기 컴퓨터공학부 박사과정  
 관심분야: Motion planning, Multi-agent system



이범희

1978 서울대학교 전자공학과 (학사)  
 1980 서울대학교 전자공학과 (석사)  
 1985 Univ. of Michigan Computer, Information & Control Eng. (Ph.D)  
 1985~1987 Purdue Univ. School of Electrical Engineering 조교수  
 1987~현재 서울대학교 전기컴퓨터공학부 교수  
 2004~현재 IEEE Robotics and Automation Society Fellow  
 관심분야: 다개체 동작계획, 센서융합, Computer