

이동 로봇의 포텐셜 함수를 이용한 장애물 회피에 관한 연구

남문호*, 김민수**, 이승환*, 박형준*, 정찬수*
 *승실대 전기공학과, **세종-록히드마틴 우주항공연구소

A Study on Obstacle Avoidance of Mobil Robot using Potential Function

*Mun-Ho Nam, **Min-Soo Kim, *Seung-Whan Lee, *Hyung-Joon Park, and *Chan-Soo Chung
 *Dept. of Electrical Eng. Soongsil Univ.

Abstract - 최근 여러 가지 자동화 서비스를 제공하기 위하여 이동로봇이 스스로 장애물을 감지하여 회피하고 이동경로를 생성할 수 있는 능력을 필요로 하고 있다.

이동 경로상에 장애물을 회피하는 방법에는 여러 가지가 있으나, 본 논문에서는 이동 경로상에 존재할 수 있는 장애물과의 충돌을 회피하여 원하는 목적지에 도달하는 방법에 대하여 연구하였다. 이를 위해 포텐셜 함수 방법을 사용하여 장애물과 목표점에 인공적인 포텐셜을 부여하여 이를 실시간 충돌 회피 문제에 적용해 보았다.

1. 서 론

공장 자동화 및 의료 분야 등 산업화로 인하여 환경변화에 유연하게 대처할 수 있는 이동 로봇(mobile robot)의 필요성이 최근 증가되고 있다. 이동 로봇은 이동을 하며 여러 가지 작업을 할 수 있기 때문에 산업계뿐만 아니라 사무실이나 가정에까지 그 응용범위가 광범위하다. 이의 한 형태인 자율주행에 관한 연구로서는 경로상에 존재할 수 있는 장애물체와의 충돌을 회피하여 최적의 경로를 구하여 찾아가는 경로 계획(path planning)에 대한 연구가 진행되고 있다.

Kant 등은 이동 장애물과의 충돌을 회피하기 위하여 Geometric Approach를 이용하여 경로를 변경하지 않고 계획된 경로 내에서만 속도를 감속하거나 가속하여서 이동 장애물과의 충돌을 회피하고자 하였다.[1]

Kyriakopoulos 등은 이동 장애물과의 거리를 추정하여 움직이는 장애물과의 충돌을 예측하는 거리함수를 만들어 충돌을 회피하는 방법을 제시하였다.[2]

Borenstein 등은 장애물의 감지에 초음파 센서를 사용하기 위해 인공 전위계 방법을 응용한 가상 역장(VFF: Virtual Force Field)과 벡터장 히스토그램(VFH: Vector Field Histogram)방법을 제안하였다.[3]

로봇분야에서 광범위하게 사용되는 또 다른 방법으로는 포텐셜 함수 방법이 있다.[4][5] 장애물은 포텐셜에 의한 척력을 발생하고 목표점은 인력을 발생시키도록 장애물과 목표점에 인공적인 포텐셜을 부여한다. 포텐셜 접근 방법에서 많이 쓰이는 회피 법칙으로는 로봇이 현재 위치에서 가장 포텐셜이 낮은 쪽을 찾아 이동하는 것이다.

본 논문에서는 포텐셜함수 방법에 의한 포텐셜력과 상대속도를 사용하여 실시간으로 구현 가능한 유도/회피 알고리즘을 연구하였다.

2. 시스템 모델

본 논문에서는 주행중인 다른 로봇이 있을 때 이를 회피하여 최종 목표점까지 유도하는 문제를 2차원 평면에서 고려하였다. 그림 1은 로봇 i가 목표점을 향해 주행 중 다른 로봇 j 근처에서 주행하는 상황을 나타낸다. 일정한 속도 V_i 로 움직이는 로봇 i의 운동방정식을 직교좌표계로 표현하면 다음과 같다.

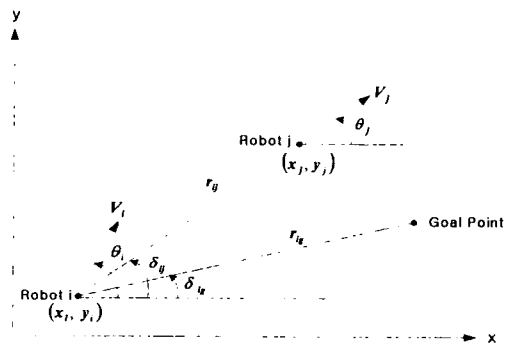


그림. 1 2차원 평면상의 형상

$$\dot{x}_i = V_i \cos \theta_i \tag{1}$$

$$\dot{y}_i = V_i \sin \theta_i \tag{2}$$

$$\dot{\theta}_i = \Gamma_i u_i, \quad |u_i| \leq 1 \tag{3}$$

여기서 (x_i, y_i) 는 위치좌표, θ_i 는 방향각, V_i 는 주행속도, Γ_i 는 최대 각가속도 크기, 그리고 u_i 는 제어입력으로 정규화된 추가속도를 나타낸다. 그림 1로부터 로봇 i, j 그리고 목표점까지의 상대거리 및 방향각의 상대적 물리량과 직교좌표와의 관계는 다음과 같이 주어진다.

$$r_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \tag{4}$$

$$r_{iy} = \sqrt{(x_{iy} - x_i)^2 + (y_{iy} - y_i)^2} \tag{5}$$

$$\delta_{ij} = \tan^{-1} \left(\frac{y_j - y_i}{x_j - x_i} \right) \tag{6}$$

$$\delta_{iy} = \tan^{-1} \left(\frac{y_{iy} - y_i}{x_{iy} - x_i} \right) \tag{7}$$

여기서 r_{ij} 는 로봇 i와 j의 상대거리, r_{iy} 는 로봇 i로부터 목표점까지의 거리, δ_{ij} 는 로봇 i에서 로봇 j를 바라보는 방향각, 그리고 δ_{iy} 는 로봇 i에서 목표점을 바라보는 방향각을 나타낸다.

로봇의 유도/회피기동은 다른 로봇과의 상대거리를 안전거리보다 크게 유지하면서 목표점에 도달하는 것으로 다음 식을 표현할 수 있다.

$$r_{ij} > r_{safe}, \quad \lim_{t \rightarrow \infty} r_{iy} = 0$$

여기서 r_{safe} 는 안전거리를 나타낸다.

3. 포텐셜 함수

포텐셜 함수 방법에서는 장애물과 목표점에 각각 척력과 인력을 발생시켜 로봇이 장애물과 목표점간의 인공적인 포텐셜장 속을 주행하도록 한다. 여기서는 다음과 같은 포텐셜 함수를 고려한다.

$$\begin{aligned} U_a(r_{ij}) &= K_a \ln r_{ij} & (8) \\ U_r(r_{ij}) &= -K_{ij} \ln r_{ij} & (9) \end{aligned}$$

여기서 U_a 는 로봇 i 의 목표점의 인력 포텐셜 함수이고, U_r 는 로봇 i 에 대한 로봇 j 의 척력 포텐셜함수를 나타낸다. 로봇 i 에 작용하는 인력과 척력은 다음과 같이 표시된다.

$$\begin{aligned} \vec{F}_{a_i} &= \nabla U_a = \frac{K_a}{r_{ij}} \vec{e}_{r_{ij}} & (10) \\ &= \frac{K_{ai}}{r_{ij}} (\cos \delta_{ij} \vec{i} + \sin \delta_{ij} \vec{j}) \\ \vec{F}_{r_j} &= \nabla U_r = \frac{K_{r_{ij}}}{r_{ij}} \vec{e}_{r_j} & (11) \\ &= \frac{K_{r_{ij}}}{r_{ij}} (\cos \delta_{ij} \vec{i} + \sin \delta_{ij} \vec{j}) \end{aligned}$$

여기서 $K_a, K_{r_{ij}}$ 는 양의 상수이며, \vec{i}, \vec{j} 는 직교좌표계의 단위벡터, $\vec{e}_{r_{ij}}, \vec{e}_{r_j}$ 는 방향의 단위벡터, r_{ij}, r_{ij} 는 각각의 상대거리, 그리고 $\vec{F}_{a_i}, \vec{F}_{r_j}$ 는 각각 로봇의 i 에 작용하는 인력과 척력벡터를 나타낸다.

4. 장애물에 대한 유도/회피

4.1 단일 이동 장애물

포텐셜함수는 위치만의 함수이므로 포텐셜력은 로봇과 목표점 또는 다른 로봇과의 상대위치에 의해서 결정된다. 그러나 이동 장애물이 있을 때 로봇의 유도/회피에서는 상대위치뿐만 아니라 상대속도 또한 중요한 요소가 되므로 포텐셜력과 상대속도를 동시에 고려하여야 한다.

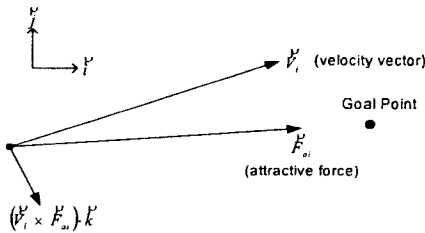


그림 2 유도 명령

그림2는 목표점에서 작용하는 인력벡터와 로봇의 속도벡터를 보여준다. 인력벡터는 목표점을 향하고 있으므로 로봇의 속도벡터의 방향이 이와 일치하면 로봇은 목표점에 도달하게 되며 속도벡터의 방향이 인력벡터와 일치하도록 다음과 같은 추가속도 명령을 고려한다.

$$a_y = (\vec{V}_i \times \vec{F}_{a_i}) \cdot \vec{K} \quad (12)$$

식 (12)로부터 유도명령은 속도벡터와 인력벡터의 외적으

로 나타나므로 두 벡터가 수직일 때 가장 큰 유도명령이 발생되며, 유도명령은 상대거리와 속도를 동시에 고려하게 된다.

속도벡터는 $\vec{V}_i = V_i \cos \theta_i \vec{i} + V_i \sin \theta_i \vec{j}$ 이므로 식 (6), (10) 과 식 (12)으로부터

$$a_y = -\frac{K_a}{r_{ij}} V_i \sin(\theta_i - \delta_{ij}) = K_a \delta_{ij} \quad (13)$$

식 (13)은 유도 추가속도 명령으로 나타낸다. 두 로봇의 거리가 안전거리 r_{safe} 내에 있을 때를 충돌로 고려할 때 그림 3은 로봇 i 와 j 의 충돌여부를 나타내는 충돌원형을 보여준다.

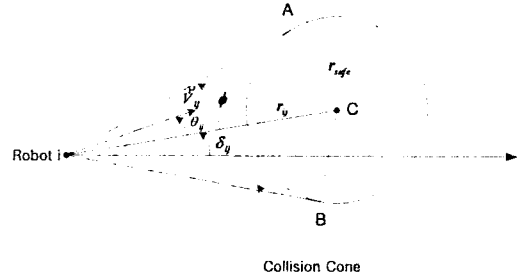


그림 3 충돌 원형

두 로봇이 등속도로 이동할 경우 그림 3의 원형 OACB내에 상대속도벡터가 위치하면 충돌이 발생하게 된다. 이 때 원추 OACB를 충돌원형이라 한다. 그림 3으로부터

$$\tan \theta = \frac{r_{safe}}{\sqrt{r_{ij}^2 - r_{safe}^2}}$$

상대속도 벡터 \vec{V}_{ij} 를 극좌표계 단위벡터로 표시하면

$$\vec{V}_{ij} = -\dot{r}_{ij} \vec{e}_r + \delta_{ij} \vec{e}_\delta \quad (14)$$

이다. 따라서 $\tan \theta_{ij} = \frac{-\dot{r}_{ij} \delta_{ij}}{r_{ij}}$ 이다. 충돌은 접근속도가 음이고 $\theta_{ij} \leq \phi$ 일 때 발생하므로, 충돌조건은 다음 식으로 주어진다.

$$\dot{r}_{ij} < 0 \text{ 이고 } \left| \frac{r_{ij} \delta_{ij}}{r_{ij}} \right| \leq \frac{r_{safe}}{\sqrt{r_{ij}^2 - r_{safe}^2}} \quad (15)$$

로봇 i 는 로봇 j 와의 상대속도가 충돌원형내에 존재하면 충돌이 예상되므로 회피기동을 수행한다. 두 로봇이 서로 가까워지고 있을 때 충돌 회피를 위해서는 상대속도벡터가 척력 벡터와 90° 를 이루도록 회피명령을 발생하는 것이 바람직하며, 다음 회피기동 명령을 고려하여야 한다.

$$a_a = -(\vec{V}_{ij} \cdot \vec{F}_{r_j}) \text{SIGN}((\vec{V}_{ij} \times \vec{F}_{r_j}) \cdot \vec{K}) \quad (16)$$

여기서 $\text{SIGN}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$

식 (16)은 회피명령의 방향은 유도명령과 반대임을 보여준다. 식 (4), (6), (11)과 (14)로부터 회피명령 (16)은 다음과 같이 주어진다.

$$a_o = \frac{K_{r_v}}{r_i} \{V_j \cos(\theta_j - \delta_{ij}) - V_i \cos(\theta_i - \delta_{ij})\} \text{SIGN}(\delta_{ij})$$

$$= K_{r_v} \frac{\dot{r}_{ij}}{r_{ij}} \text{SIGN}(\delta_{ij}) = -K_{r_v} t_{go} \text{SING}(\delta_{ij}) \quad (17)$$

식 (17)로부터 회피 명령의 방향은 유도 증가속도 명령의 반대방향으로 나타나고, 크기는 예상되는 장애물까지의 도달시간에 비례한다.

로봇이 충돌경로상에 있지 않을 때 회피기동을 할 필요가 없으나 이 경우 상대속도벡터가 충돌원추의 경계, \overline{OA} 또는 \overline{OB} 에 있을 때, 회피명령의 불연속이 발생한다. 불연속성을 없애기 위해 다음 상수

$$K_{conc} = \begin{cases} 0 & : \theta_{ij} > 1.1\phi \\ -10(\theta_{ij} - 1.1\phi) & : 1.1\phi \geq \theta_{ij} \geq \phi \\ 1 & : \phi > \theta_{ij} \end{cases}$$

를 식 (17)에 곱하여 다음과 같이 최종 회피가 속도 명령을 구한다.

$$a_o = -K_{conc} K_{r_v} t_{go} \text{SING}(\delta_{ij}) \quad (18)$$

로봇의 유도/회피 기동은 다음과 같이 이루어진다.

$a_o \neq 0$ 일 때

a_o 와 a_g 가 같은 방향이면

$$a_c = \max(a_o, a_g)$$

아니면

$$a_c = a_o$$

$a_o = 0$ 일 때

$$a_c = a_g$$

$$u_i = \begin{cases} a_c/a_{\max} & : |a_c/a_{\max}| < 1 \\ \text{SIGN}(a_c/a_{\max}) & : \text{otherwise} \end{cases}$$

4.2 다수의 이동 장애물

다수의 이동장애물이 있을 때는 식 (18)에 따라 각 장애물에 대한 회피 기동 명령을 구한다. 가속도 명령의 크기는 회피의 우선순위를 나타냄으로 가장 큰 가속도 명령을 필요로 하는 장애물부터 회피를 하며 주행한다.

이 방법에서 장애물은 등속으로 이동하는 것으로 가정하였으므로 장애물간의 간격이 좁을 경우 그 사이를 지나가는 것은 위험할 수 있다. 따라서 매우 좁은 간격으로 모여 있는 장애물들은 군집화를 시켜 군집사이를 이동하는 것을 회피하는 것이 바람직하다.

로봇 i 의 관점에서 이동 장애물인 로봇 j 와 k 사이의 간격은 다음과 같이 주어진다.

$$r_{jk} = \sqrt{r_{ij}^2 + r_{ik}^2 - 2r_{ij}r_{ik} \cos(\delta_{ij} - \delta_{ik})} \quad (19)$$

로봇 j 와 군집을 이루는 군집장애물은 서로간의 간격이 일정거리 d 이내인 장애물의 집합으로 다음과 같이 정의된다.

$$C = \{Aircraft \ k \mid r_{jk} < d, k \neq j, k \neq i\} \quad (20)$$

여기서 C 는 군집장애물 집합을 나타낸다. 군집 장애물은 전체의 척력과 속도로 회피기동을 계산하며, 군집장애물의 척력은 군집을 이루는 각 장애물 척력의 합으로 다음과 같이 정의된다.

$$\vec{F}_{cluster} = \sum_{k \in C} \vec{F}_{r_k} \quad (21)$$

군집장애물에 대한 상대속도는 최악의 경우를 생각하여 군집 중 가장 큰 회피기동을 요구하는 장애물에 대한 상대속도로 가정한다. 단일 장애물에 대한 회피기동인 식 (16)과 유사하게 군집장애물에 대한 회피기동 명령은 다음과 같이 계산한다.

$$a_u = -(\vec{V}_{ij} \cdot \vec{F}_{cluster}) \text{SIGN}((\vec{V}_i \times \vec{F}_{cluster}) \cdot \vec{K}) \quad (22)$$

다수의 이동장애물이 있을 때 로봇의 유도/회피 기동은 다음과 같이 이루어진다.

1. 각 이동 장애물에 대해 회피 증가속도 계산
2. 가장 큰 회피 증가속도를 요구하는 장애물인 로봇 j 를 결정
3. 로봇 j 가 군집 장애물인지 판단
4. 군집장애물이면
식 (21)로 회피 증가속도 계산
아니면
식 (18)로 회피 증가속도 계산
5. 식 (13)로 유도 증가속도 계산
6. $a_o \neq 0$ 일 때

a_o 와 a_g 가 같은 방향이면

$$a_c = \max(a_o, a_g)$$

아니면

$$a_c = a_o$$

$a_o = 0$ 일 때

$$a_c = a_g$$

7. 제어명령 계산

$$u_i = \begin{cases} a_c/a_{\max} & : |a_c/a_{\max}| < 1 \\ \text{SIGN}(a_c/a_{\max}) & : \text{otherwise} \end{cases}$$

5. 결 론

본 논문은 자율주행을 위한 유도/회피 법칙을 유도하였다. 이동 장애물에서는 척력을 목표점에는 인력을 인공적으로 부여하여 로봇의 척력과 인력이 작용하도록 하고 이를 포텐셜력과 상대속도를 사용하여 실시간으로 구현 가능한 유도/회피법칙을 구현하였다. 추후 연구과제로 본 논문에서 제안한 방법을 직접 시뮬레이션 및 실험을 통하여 유용성을 평가해 보아야 할 것이다.

[참 고 문 헌]

- [1] I. K. Kant and S. W. Zuer, "Toward efficient trajectory planning : the path-velocity decomposition", The International Journal of Robotic Research, vol. 5, no. 3, pp. 72-89, 1986
- [2] K. J. Kyriakopolos and G. N. Saridis, "An integrated collision prediction and avoidance scheme for mobile robots in non-stationary environments", Automatica, vol. 29, no. 2, pp. 309-322, 1993.
- [3] J. Borenstein and Y. Koren, "The vector field histogram fast obstacle avoidance for mobile robots", IEEE transactions on Robotics and Automation, vol. 7, no. 3, pp. 278-287, 1991.
- [4] T. Balch and M. Hybinette, "Social potentials for scalable multi-robot formations", IEEE international on Robotics and Automation, vol. 1, no. 24-28, pp. 73-80, 2000.
- [5] S. S. Ge and Y. J. Cui, "New potential functions for mobile robot path planning", IEEE transactions on Robotics and Automation, vol. 16, no. 5, pp.615-620, 2000.