

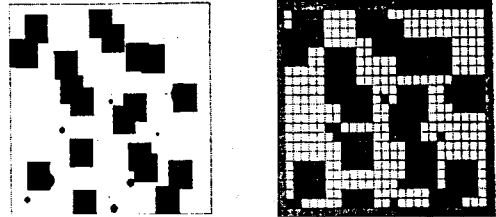
다중이동로봇의 장애물 회피 논리 및 경로계획에 관한 연구

박경진* . 이기성 . 이종수
 홍익대학교 전기제어공학과

A Study on the Obstacle Avoidance Algorithm and Path Planning of Multiple Mobile Robot

Kyung-Jin Park* , Ki-Sung Lee , Jong-Soo Lee
 Hong-Ik University, Dept. of Electrical & Control Eng.

Abstract - In this paper, we propose a new method of path planning for multiple mobile robot in dynamic environment. To search the optimal path, multiple mobile robot is always generating path with static and dynamic obstacles avoidance from start point to goal point. The purpose of this paper is to design an optimal path for the mobile robot.



(a) 집합 map (b) 집합 MAP

1. 서 론

현재 산업현장에서 응용되는 대부분의 로봇방식은 매니플레이터방식으로, 작업공간의 한계성이 있었다. 이러한 단점을 보완하기 위하여 자율주행로봇방식이 필요하게 되었다. 최근에는 한 대의 로봇으로 해결하기 어려운 문제를 여러 이동로봇으로 해결하는 방식이 대두되고 있으며, Service Robot의 비중의 증대로 동적환경하에서의 경로계획에 대한 필요성이 커져가고 있다. 이를 위해, 본 논문에서는 위와 같은 점에 주안하여, 이동로봇을 고정된, 움직이는(이동로봇)장애물이 있는 환경하에서 장애물 회피시마다 최단경로를 생성하여 주어진 목표점까지 최단거리로 이동시키는 경로계획에 관하여 연구하였다. 본 논문의 경로계획 과정은 셀분해법(Cell Decomposition)을 거친 후 거리변환법(Distance Transform Method)을 적용하는 단계로 구현하였고, 가상 장애물(Virtual Obstacle) 개념을 도입하여 움직이는 장애물(이동로봇)을 회피할 수 있도록 하였다. 또한, 유선화 과정(Smoothing)을 거리변환(DT)으로 구한 곡선에 적용시켜 보다 더 부드러운 경로곡선을 구하였다. 계획하고 있는 환경을 크게 두 부류로 나누면, 1)여러대의 로봇과 고정된 장애물들이 있는 경우 2)여러대의 로봇과 여러대의 움직이는 장애물들이 있는 경우로 나눌 수 있겠다.

2. 본 론

2.1 셀 분해법(Cell Decomposition)

경로계획을 하는 과정에서, 활동영역을 묘사하기 위한 과정으로 본 논문에서는 셀 분해법을 적용하였다. 셀 분해법은 활동영역을 셀 단위로 처리하기 때문에 활동영역에 대한 데이터가 줄어들고 알고리즘이 간단하며 반복적으로 구현할 수 있다는 장점이 있다. 본 논문에서는 한 셀을 로봇이 통과할 수 있는 크기로 나누었고, 한 셀의 면적의 10%이상을 장애물이 차지 할 경우는 장애물이 존재하는 셀로 설정하였다. 또한, 장애물이 있는 셀은 '1'로 그렇지 않은 경우는 '0'으로 설정하였다.

$$map(x, y) = \begin{cases} 1: obstacle \\ 0: otherwise \end{cases}$$

지도로부터 입력된 집합 map을 셀 단위로 분해된 집합 MAP으로 변환한다.

$$C_{DIVISION} \cdot map \rightarrow MAP$$

그림 1. 셀 분해법

$$MAP = [M: M(x, y), x=0 \sim W_{MAP}-1, y=0 \sim H_{MAP}-1]$$

$$W_{MAP} = \frac{W_{map}}{CWH}, \quad H_{MAP} = \frac{H_{map}}{CWH}$$

$$CWH = \max(W_R, H_R)$$

$$M(x, y) = \begin{cases} 0: O_{PA} < 10 \\ 1: x=0 \text{ or } y=0 \text{ or } x=W_{MAP}-1 \\ \text{ or } y=H_{MAP}-1 \\ 2: O_{PA} \geq 10 \end{cases}$$

$$O_{PA} = \frac{A_{OB}}{A_{CELL}} \times 100$$

$$A_{OB} = \sum_{x=0}^{CWH-1} \sum_{y=0}^{CWH-1} map(x^*W_{MAP} + X, y^*H_{MAP} + Y)$$

$$A_{CELL} = CWH^2$$

W_{MAP}, H_{MAP} 은 셀분해된 지도 MAP의 행과 열의 개수를 의미하며, W_{map}, H_{map} 은 각각 제공된 지도 픽셀의 행과 열의 개수를 의미한다. W_R 과 H_R 은 로봇의 폭과 높이를 의미하며, CWH 는 로봇의 폭 또는 높이 중 큰 값을 의미한다. A_{OB} 는 장애물의 면적이고, A_{CELL} 은 셀의 넓이이다.

2.2 DT(Distance Transform)

셀 분해법으로 나누어진 활동 영역에서 최적경로를 찾기 위한 탐색 알고리즘으로는 깊이 우선 탐색(Depth First Search), A*, 거리변환(Distance Transform) 알고리즘 등이 있다. 이중 거리변환법은 숫자를 확산시키며 최단거리 경로를 찾는 알고리즘으로 다른 탐색 알고리즘에 비해 최적경로를 보장하고, 프로그램으로 구현하기가 간단하다는 장점을 가지고 있다.

본 논문에서 사용할 DT 알고리즘을 살펴보자.

- 1) 격자화된 C-space와 같은 크기의 메모리 공간을 만든다.
- 2) 장애물이 점유된 셀에 해당하는 메모리에는 입력이 가능한 가장 큰 수를 입력시키고 도착점에는 1을 입력하고 나머지 부분에는 C-space의 전체 셀의 개

수를 입력한다.

3) 메모리의 한 셀씩을 주변의 4-neighbor or 8-neighbor와 비교하며 자신의 숫자 보다 작은 숫자가 있으면 그 숫자에 1을 더해 자신의 메모리 셀을 갱신한다. 단, 장애물일 때는 다음 셀로 넘어간다.

이 과정을 식으로 표현하면 다음과 같다.

(4-neighbor인 경우)

if $(C(x,y) \neq \text{장애물})$

$$C(x,y) = \min(C(x+1,y)+1, C(x-1,y)+1, C(x,y+1)+1, C(x,y-1)+1, C(x,y))$$

4) 숫자의 변경이 출발점에 해당하는 메모리 셀에서 일어난 것을 검사하고 일어나지 않았다면 3)의 과정으로 돌아간다.

5) 출발점에서부터 1씩 작은 숫자를 따라 도착점까지 도달하여 최단 경로를 결정한다.

이상의 과정을 그림 2에 나타내었다.

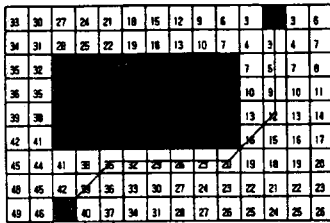
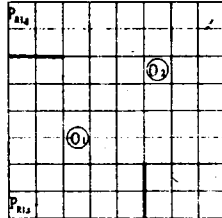
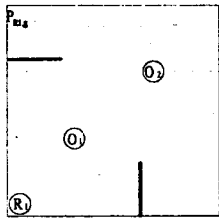
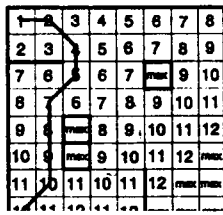
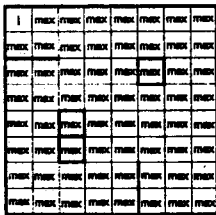


그림 2. Distance Transform(DT)



<활동 영역상의 로봇과 장애물 위치>

<셀분해법으로 활동영역을 나눈 모습>



<장애물 위치표시 및 메모리공간 초기화>

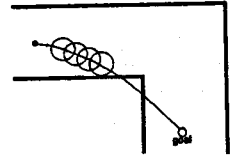
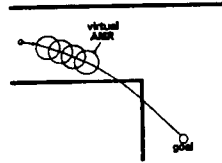
<DT방법으로 구한 최단 경로>

그림 3. 셀분해법과 DT를 이용해 경로를 구하는 과정

2.3 가상 장애물(Virtual Obstacle)

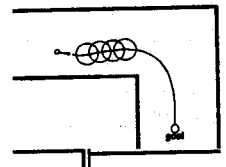
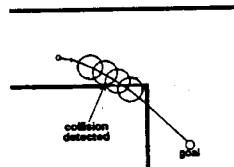
움직이는 장애물을 고려한 경로를 구하고, 주행 중 경로를 수정하는 시기를 결정하기 위해서 본 논문에서는 가상 AMR(Autonomous Mobile Robot)을 제안하였다. AMR이 경로를 따라 주행할 때 n step 이후 AMR이 경로 상에 위치할 곳에 가상 AMR을 위치 시켜보아 장애물과 충돌이 발생하는 지 여부를 검사하여 충돌이 발생한다면 그 시점에서 새로운 경로를 찾는다. 가상 AMR의 형태는 처리상의 편의를 위해 원형의 모델로 정의하고 그 크기는 실제의 AMR보다 크게 정하여 필요한

만큼의 안전도를 얻었다. 몇 step 이후를 검사할 것인지는 Simulation이나 실험을 통하여 정할 수 있는데, 몇 step 이후 한 위치에 대해서만 충돌 여부를 검사하는 것보다는 몇 step 이후부터 얼마정도의 범위를 가상 AMR을 미리 보내 검사하는 것이 유리하다. 그림 4에서는 가상 AMR을 도입하였을 때의 AMR의 경로를 보여준다. (a)에서 가상 AMR을 도입하여 (b)에서 AMR은 가상 AMR을 2step 앞에서 5step 앞까지 보내서 벽과의 충돌 여부를 검사하지만 아직 장애물이 인식되지 않은 관계로 가상 AMR과 벽과의 충돌은 일어나지 않았다. (c)에서 AMR이 장애물과 가까워짐에 따라 센서로 장애물을 인식하게 되었고 가상 AMR의 충돌이 감지되었다. (d)에서 AMR은 새로운 경로를 찾아 그 경로를 따라 주행하는 것을 볼 수 있다.



<그림 a>

<그림 b>



<그림 c>

<그림 d>

그림 4. 가상 AMR을 이용한 경로

2.4 경로의 유선화(Smoothing)

본 논문에서는, 구한 경로를 유선화(Smoothing)하여 방향전환이 완만하게 이루어 지도록 유선화를 통해 얻어진 새로운 점들은 로봇이 주행하는데 목표점으로 사용하였다.

$P = \{P_i(x,y) | i=1 \text{ to } n, n=\text{경로를 구성하는 셀의 개수}\}$

여기서, P를 유선화하여 얻은 집합 S는

$$S_{2j-1}(x,y) = \frac{TR_{2j-1}(x,y) + (T-\tau)P_{2j}(x,y)}{\tau}$$

$$S_{2j}(x,y) = \frac{(T-\tau)P_{2j-1}(x,y) + TP_{2j}(x,y)}{\tau}$$

$$\text{단, } 0 < T < \tau, T > \frac{\tau}{2}$$

$$S = \{S_j(x,y) | j=1 \text{ to } 2 \times n\}$$

유선화과정에서 고려할 사항은, 위 두식에서 구한 경로를 유선화하는데 있어 로봇의 위치를 로봇 중심 한점으로 보기 때문에 로봇이 셀과 셀의 경계면을 지나면 로봇이 위치한 셀이 변경된다. 따라서 경로를 구하게 되면 $P_1(x,y)$ 자리에는 다음 셀의 중심좌표가 들어가게 된다. 이를 유선화하면 현재 로봇이 위치한 지점은 유선화된 경로상에 포함이 되지 않게 된다. 이를 해결하기 위해 집합 P를 수정하여 유선화과정을 거칠 때 로봇의 위치가 항상 포함되도록 해야한다.

따라서,

$$P(t_m) = \begin{cases} [P_i(x, y)|i=1 \sim n]: \text{단, } P_1(x, y) = P_{R_i}(t_m) \text{ 일 때} \\ [P_i(x, y)|i=0 \sim n] \\ : \text{단, } P_1(x, y) \neq P_{R_i}(t_m) \text{ 일 때, } P_0(x, y) = P_{R_i}(t_m) \end{cases}$$

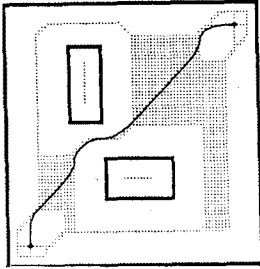


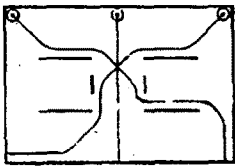
그림 5. DT 경로 탐색 후 유선화과정을 거친 경로곡선

3. 결 론

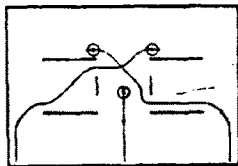
본 논문은 셀분해법, DT, DT를 거친 후의 유선화과정 및 움직이는 장애물을 회피하기 위하여 가상장애물을 도입하여 여러 이동로봇이 동적환경하에서 장애물을 회피하여 목표점까지 최적경로로 도달할 수 있는 알고리즘을 제안하였다. 제안한 알고리즘으로 Simulation을 통하여 동적환경하에서 장애물과의 충돌없이 최적경로를 생성하는 것을 확인 하였다.

[참 고 문 헌]

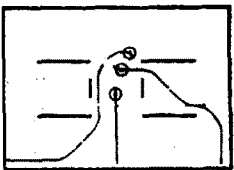
- [1]김상문, "거리변환과 VFH를 이용한 이동로봇의 장애물 회피", 홍익대학교 석사학위논문, 1997
- [2]곽한택, "유전알고리즘을 이용한 자율주행로봇의 장애물 회피", 홍익대학교 석사학위논문, 1996
- [3]홍순학, "알려진 환경에서 자율주행로봇의 알려지지 않은 이동 장애물 회피논리", 한양대학교 석사학위논문, 1995
- [4]김선옥, "움직이는 장애물이 있을 때 이동로봇의 충돌회피 알고리즘에 관한 연구", 한국과학기술원 석사학위논문, 1994
- [5]정현, "유전알고리즘을 이용한 이동로봇의 주행알고리즘 개발", Trans. KIEE, VOL. 48A, NO.7, JUL 1999
- [6]Philip John Mckerrow, "Introduction to Robotics", Addison Wesley, 1991
- [7]T.Arai and J.Ota, "Motion Planning of Multiple Mobile Robots", Proc. IEEE International Conference on Intelligent Robots and Automation, pp.1761-1768, 1992
- [8]M.Erdmann, Lozano-Perez, T., "On Multiple Moving Objects", Proc. IEEE International Conference on Robotics and Automation, pp.1419-1424, 1986



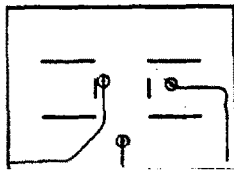
(a) 단계1(출발)



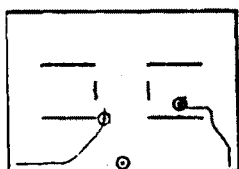
(b) 단계2



(c) 단계3



(d) 단계4



(e) 단계5



(f) 단계6(종료)

그림 6. 여러대의 이동로봇의 경로계획