

mobile robot의 장애물 회피방법 개선

박종훈, 이우영, 허대정, 허욱열
인하대학교 전기공학과
전화 : 032-860-7394 / 핸드폰 : 019-634-1164

Improving on the Obstacle Avoidance Method for a Mobile Robot

Jong-Hun Park, Woo Young Lee, Dei Jeung Huh, Uk Youl Huh
Dept. of Electrical Engineering, Inha University
E-mail : patch_hanl@hotmail.com

Abstract

This paper presents collision avoidance for mobile robots equipped with synchro-drive using curvature trajectory by the obstacle type. The new real-time obstacle avoidance method presents how to create a curvature trajectory in which dynamics of a mobile robot is considered we controlled translation and rotational velocity of the mobile robot. Using these two speeds with curvature trajectory, the mobile robot navigates to target point without collision. We consider that the robot going to curvature trajectory by obstacle size towards a goal location. The collision avoidance has been implemented and tested using pioneer2-dxe mobile robot.

I. 서론

실내에서 이동하는 지능형 자율이동 로봇의 요구조건은 목표로 하는 목적지까지 안전하고 빠르게 주행해야만 한다[3]. 예측할 수 없는 환경 속에서 로봇의 행동은 발생한 문제에 대하여 신속하게 처리하며 본래 하고 있는 일을 수행해야 한다. 하나의 예로써, 목표로 하는 점으로 주행 시 로봇이 인지하고 있지 않은 장애물이 존재할 경우, 장애물을 발견하고 안전하고 신속하게 회피하여 목표로 하는 점으로 이동한다[4]. 신속

하게 장애물을 회피 할뿐 아니라, 이동로봇을 가속구성원 중의 하나로 친밀도를 높일 수 있는 지능적인 행을 요구되어진다. 장애물을 만났을 때 부드럽게 원 궤적을 그리면 선회하는 함으로써 인간 아닌 다른 물체의 움직임에 좀 더 친밀함을 전해준다. 이러한 장애물 회피방법에 있어서 global 장애물 회피 방법과 local 장애물 회피 방법이 있다. 첫 번째 방법은 로봇의 초기위치에서 목표위치까지 회피경로를 완벽하게 연산하여 얻어지는 장애물인지 방법이다. 이 방법은 off-line 상에서 이루어지므로 계산과정에 너무 많은 시간을 소비하게 되어지고, 로봇이 인지하지 못하고 있는 환경에서의 장애물회피에 적합하지 않다. 후자의 방법은 장점은 실내 환경의 정보가 계속 바뀌는 상황에서 바뀐 환경에 신속한 계산과 환경에 대한 적응이 가능하다. 하지만 최적의 경로를 만들지 못하고 지역 최소점에 빠진다는 단점을 가지고 있다[5][6]. 대부분 장애물을 회피 시 local 장애물 회피방법을 사용한다. 본 논문에서 사용하고 있는 방법은 후자의 방법이다. 곡선 궤적을 이용한 장애물 회피 방법은 신속하고 정확하게 장애물을 회피한다. 하지만 센서 정보의 부정확성과 로봇의 거리정보를 가져오는 주기가 맞지 않을 경우 장애물과 너무 가까운 위치에서 궤적을 생성하게 되고 그로 인하여 충돌이 일어난다. 또한 곡선 궤적을 이용하여 회피할 때 장애물의 형태를 고려하지 않았다.

이러한 문제를 개선하기 위하여 원 궤적을 3개를 만들어 주고, 궤적을 따라 회피할 때, 로봇과 장애물 사이의 거리를 고려하여 안전한 새로운 궤적을 만들어 줌으로써 좀더 안정적인 회피방법을 제안하였다.

II. 근사화된 운동방정식과 곡선궤적

2.1 Synchro-Drive Robot의 운동방정식

본 논문에서는 synchro-drive 로봇의 운동방정식을 이용하여 이동속도와 각속도를 독립적으로 제어할 수 있는 간략화된 운동방정식을 사용하였다[1][5]. 식 (1)과(2)는 실제 로봇에 적용시키기 위해 간략화된 식들이다.

$$x(t_n) = x(t_0) + \sum_{i=0}^{n-1} (H_x^i(t_{i+1})) \quad (1)$$

$$H_x^i(t) = \begin{cases} \frac{v_i}{\omega_i} (\sin \theta(t_i) - \sin(\theta(t_i) + \omega_i(t - t_i))), \omega_i \neq 0 \\ v_i \cos(\theta(t_i)) \cdot t, \omega_i = 0 \end{cases}$$

$$y(t_n) = y(t_0) + \sum_{i=0}^{n-1} (H_y^i(t_{i+1})) \quad (2)$$

$$H_y^i(t) = \begin{cases} -\frac{v_i}{\omega_i} (\cos \theta(t_i) - \cos(\theta(t_i) + \omega_i(t - t_i))), \omega_i \neq 0 \\ v_i \sin(\theta(t_i)) \cdot t, \omega_i = 0 \end{cases}$$

식 (1)과(2)에서 각속도가 0인 경우는 ($\omega_i=0$) 직선 운동을 하는 궤적을, 그렇지 않은 경우는 ($\omega_i \neq 0$) 곡선 궤적을 따라 이동한다. 또한, M_x^i 와 M_y^i 를 식(3)과 같이 정의하면, 식(4)을 얻을 수 있게 된다.

$$\begin{aligned} C_x^i &= \frac{v_i}{\omega_i} \cdot \sin \theta(t_i) \\ C_y^i &= \frac{v_i}{\omega_i} \cdot \cos \theta(t_i) \end{aligned} \quad (3)$$

$$(H_x^i - C_x^i)^2 + (H_y^i - C_y^i)^2 = \left(\frac{v_i}{\omega_i}\right)^2 \quad (4)$$

식(4)는 좌표 점을 (M_x^i, M_y^i) 이고, 반지름이 (v_i/ω_i)=r인 원의 방정식을 표현하였다. 위의 식을 이용해서 로봇의 이동속도와 각속도를 각각 제어함으로써 원하는 로봇의 원 궤적을 만들 수 있다.

$$\omega = \frac{v}{r}, \quad t = \theta / \omega \quad (5)$$

장애물을 회피할 때 바닥과 로봇사이의 미끄러짐과 관성의 영향을 줄이기 위해 궤적을 일정한 속도와 각속도를 이용하여 시간 t 동안에 반지름이 r 인 원 궤적을 θ 만큼 선회하게 된다[9],[그림 1(a)]

2.2 곡선궤적을 이용한 장애물 회피의 문제점

앞장에서 근사화된 운동방정식의 이동속도와 각속도를 제어함으로써 원 궤적을 만들 수 있음을 알았다. 기존의 곡선궤적방법은 장애물 회피에는 큰 영향을 주지 않는 범위의 오차와, 장애물 회피 과정에서도 지속적으로 목표방향을 지향하면서 이동하였다[8]. [그림 1(a)] 이러한 장점에도 불구하고 갑작스런 장애물의 변화에 민감히 반응하지 못하고, 장애물의 형태를 고려하지 못했기 때문에 충돌 위험을 내포하고 있는 주행이었다. 이러한 단점을 개선하기 위해서는 장애물의 형태를 고려해야하며, 좀더 안전하고 빠르게 주행할 수 있도록 하여야한다. [그림 1(b)]에서는 로봇이 목표점을 따라 주행할 때 인지하고 있지 않던 장애물이 존재할 경우 곡선궤적을 만들어 회피하는 모습을 보여준다. 하지만 로봇이 장애물과의 거리를 계산하여 만들어진 궤적이 장애물의 크기를 고려하지 못한 채 이동하다가 충돌하는 모습을 보여준다. 또한 장애물을 늦게 발견하거나 센서 정보의 부제로 궤적을 만들어 회피하기에 충분한 궤적을 만들지 못할 경우도 생긴다. 이 경우는 장애물과 정면으로 충돌한다. 좀더 안정적인 궤적의 회피가 요구되어진다.

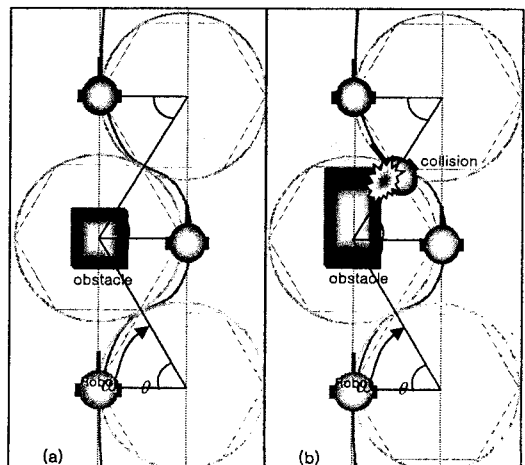


그림 1 곡선궤적을 이용한 장애물회피의 문제점

III. 장애물회피 알고리즘

3.1 회피알고리즘

본 논문에서는 [그림 2]에서와 같이 곡선궤적을 이용한 장애물회피 알고리즘을 이용하여 안전하고 신속하게 회피하도록 하였다. 초기화되어진 로봇의 주행은 장애물이 주행하고자하는 통로에 존재할 경우, 양쪽 벽과 장애물사이의 공간이 로봇이 회피하기 위한 충분한 여유 공간이 되는지 확인한다.[8] 예를 들어 장애물의 오른쪽 공간이 로봇이 안전하게 주행하기에 충분한 공간임을 확인하면 오른쪽으로 곡선궤적을 생성시켜서 이동한다. 장애물과 로봇의 사이가 너무 가까워 궤적을 만들 수 없을 경우 주행을 멈추고 궤적을 만들기 위해 충분한 거리만큼 장애물로부터 떨어져서 다시 주행을 시작한다. 원 궤적 3개를 이용하여 총 4번의 곡선궤적을 따라 주행한다. 곡선궤적을 이용하여 빠르고 안전하게 명령을 수행하는 알고리즘을 제안하였다.

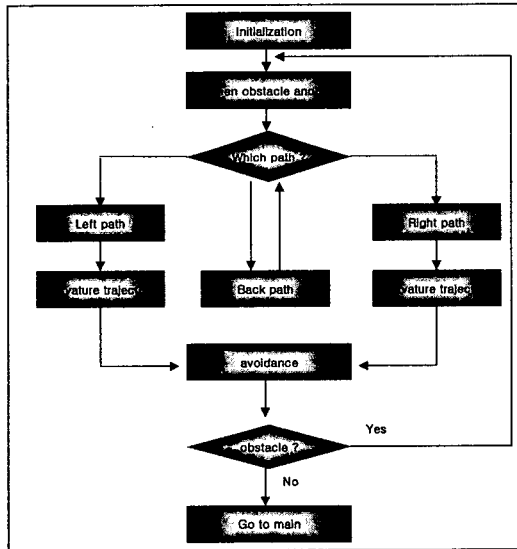


그림 2 곡선궤적을 이용한 장애물 회피 알고리즘

3.2 곡선궤적을 이용한 장애물회피방법

곡선궤적을 이용한 장애물 회피방법은 그림 2 에서와 같은 알고리즘을 이용하여 회피한다. 또한 곡선궤적을 만들고 주행할 때 문제시되었던 점을 보완함으로써 갑작스런 장애물의 출현이 있을 때 좀더 안전하고 빠르게 회피할 수 있다. 하지만 실험을 위하여 몇 가지 제한 조건이 첨가되어졌다. 우선 로봇이 이동하려는 방

향에 충분한 공간이 보장되어야하며 급격한 이동장애물의 방해가 없어야한다. 그림3(a)의 ①,③은 동일한 반지름과 각속도를 만들어 회전을 한다. 하지만 ②의 영역에서 장애물의 크기를 고려하는 방법을 제시하여 충돌 없이 안전하게 회피하는 모습을 보여주고 있다. 로봇은 고정된 속도로 주행하고 각속도를 0으로 하여 장애물이 존재하는 구간까지 로봇이 목표로 하는 방향 속도를 이용하여 장애물의 존재여부에 따라 장애물을 통과하고 장애물이 존재하지 않으면 3번째 궤적을 따라 주행한다.

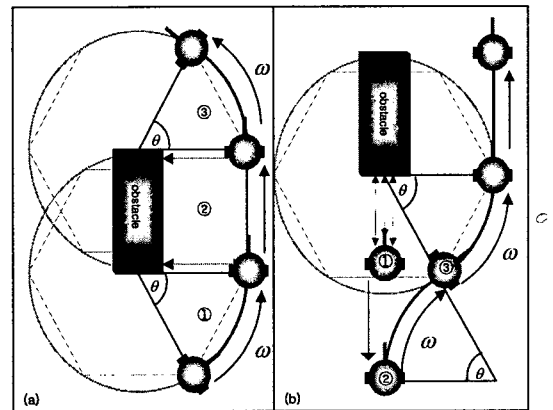


그림 3 개선되어진 장애물회피

①=③ $w = v/r, t = \theta/w$: 원 궤적 생성

② $w = 0, t_{obstacle}$: 장애물이 존재하는 구간

또한 그림3(b)는 ①의 위치에서 장애물을 발견했다고 가정할 때 궤적을 만들 수 없을 것을 확인하고 안전하게 궤적을 따라 회피할 수 있도록 ②의 궤적 초기 점으로 이동하여 회피하기에 충분하지 확인 후에 장애물 회피 알고리즘을 이용하여 안전하게 회피한다.

IV. 실험 및 결과

그림4는 일정속도에서 로봇이 장애물을 만났을 때 안전하게 회피하는 모습을 보여준다. 본래의 곡선궤적의 방법과 비교하여 도착점에서의 목표의 방향 오차는 거의 동일한 반면 장애물에 대한 좀더 안정적이고 충돌에 대한 위험부담을 줄임으로써 성공적으로 장애물회피를 실행함을 확인하였다. 로봇은 Active Media사의 Pioneer2-DXE를 사용하였으며, 센서로서는 전방에 설치된 초음파 센서를 사용하였다. 로봇의 내부적 cycle

time은 100ms이고, 센서 하나가 정보를 얻는데 걸리는 시간은 25Hz이므로, 총 8개의 센서 정보를 얻는데 걸리는 시간은 320ms가 된다. 이 실험에서는 algorithm의 cycle time을 400ms로 정하고 실험을 하였다. algorithm의 구현은 visual C++를 이용하였으며, PC에 serial port와 RS232 port를 이용하여 로봇과 통신하였다.

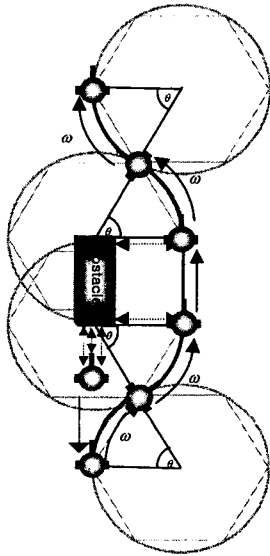


그림 4 곡선궤적을 이용한 장애물회피

V. 결 론

본 논문에서는 간략화 되어진 운동방정식을 이용하여 실제로봇에 적용하였고 기존의 곡선궤적을 이용한 장애물회피방법을 개선하여 좀더 안정적이고 충돌에 대한 위험부담을 줄이는데 초점을 두었다. 실험결과 갑작스런 장애물 위치변화와 장애물의 크기에 대한 로봇의 안정적인 회피가 이루어졌다. 하지만 본 논문에서 제안한 방법은 비록 안정적으로 장애물을 회피하는 방법이지만 회피한 후 에러의 증가가 장애물의 갑작스런 출현 빈도에 따라 목표로 하던 점의 위치오차와 방향오차가 증가될 수 있음을 실험을 통하여 확인하였다. 이러한 문제점은 위치 추정에 관한 연구가 계속되어야 겠고 이동장애물에 대한 로봇의 대응 또한 연구가 이루어져야 하겠다.

[참 고 문 헌]

- [1] Dieter Fox, "The Dynamic Window Approach to Collision Avoidance" IEEE Robotics&Automation Magazine, March 1977
- [2] L.Feng, "Cross-Coupling Motion Controller for Mobile Robots" IEEE Control Systems Magazine, Volume: 13 Issue: 6, pp. 35 - 43 Dec. 1993
- [3] Hyun S.Yang, "Integration of Topology Map and Behaviors for Efficient Mobile Robot Navigation", The 8th International Symposium of Robotics Research, 1997
- [4] Teruko YATA, "Wall Following Using Angle Information Measured by a single Ultrasonic Transducer", Proceedings of the 1998 IEEE International Conference on Robotics & Automation, pp. 1590 - 1596, May 1998
- [5] Dieter Fox, "A Hybrid Collision Avoidance Method For Mobile Robots", In Proc. of IEEE International Conference on Robotics and Automation, 1998
- [6] M.Khatib, "Indoor Navigation with Uncertainty using Sensor-Based Motions", IEEE International Conference on Robotics and Automation, April 1997
- [7] 오영선, "퍼지 논리 제어를 이용한 이동 로봇의 자율 주행에 관한 연구", 인하대학교 산업기술대학원 자동화공학과 석사 학위논문, 1996
- [8] 정갑균, "광유도식 무인반송차의 경로 추종에 관한 연구", 인하대학교 전기공학과 대학원 석사 학위 논문, 1999
- [9] 이우영 "곡선 궤적을 이용한 mobile robot의 장애물회피", 인하대학교 전기공학과, KIEE2002