

Bezier 곡선을 이용한 이륜 구동 로봇의 경로 생성

조규상
 동양대학교 컴퓨터 공학부

Trajectory Generation of Two Wheeled Mobile Robot using Bezier Curve

Cho, GyuSang
 School of Computer Eng., DongYang Univ.

Abstract

This paper describe a trajectory generation method for a two-wheeled mobile robot using cubic Bezier curve. It is proposed that the method to determine the location of control points which mainly affect the shape of curve, and constrains for two-wheeled mobile are examined. Simulation results show its traceability of the trajectory of mobile robot.

1. 서 론

이동로봇이 미지의 환경을 주행할 때, 빠르게 목표지점을 찾아가기 위해서는 이동할 경로를 적절하게 선택할 수 있어야 한다. 간단한 곡선을 위한 여러 종류의 경로 설계 방법들이 제안되었는데, 직선, 원호, clothoid와 그것들의 혼합된 형태로 사용되었다[1]. 원하는 곡선을 따라서 중량이 있는 로봇이 이동해야 하기 때문에 부드러운 곡선을 형성하는 경로는 중요한 문제이다. 이런 경우에는 경로상에 불연속점이 존재할 수 있다. 이런 경우의 곡선에서는 미끄러짐이나 편차문제등이 발생할 수 있다.

로봇이 이런 경로상을 주행할 때는 정지한 후에 다시 방향을 조정하고 이동하는 방식을 취하게 된다. 이런 경우에 대해 여러 연구가 수행되었는데, Nelsen[2]은 기구학적인 제약이 있는 차륜형 로봇의 연속 곡률을 갖는 경로에 대한 연구를 수행하였으며, 이와 유사한 연구가 Kanayama[3] 등에 의해서도 수행되었다.

경로를 생성하는 방법중에서 스플라인 형태의 곡선을 이용한 연구는 Lin등[4]에 의해서이다. Arakawa등[5]은 로봇에 장착한 비전시스템에 의해서 얻은 데이터를 이용하여 경로를 위한 위치를 구하여 그 점들을 Bezier 곡선에 의해서 연결하고 추종하는 방법을 제안하였다. Bezier와 유사한 개념의 방법으로써 B-spline을 이용한 여러 가지 방법들이 제안되었는데 Komoriya[6] 등은 B-spline곡선에 의해서 생성된 경로가 이륜구동형 이동로봇의 메카니즘으로 추종이 가능한 것인가를 곡선의 곡률 데이터를 이용하여 하드웨어의 제약이 있는 로봇의 최소 이동시간을 위한 속도 패턴을 정하고 검사하는 방법의 유용성을 실험을 통하여 입증하였다. 최근에 Paulos[7]는 B-spline 곡선을 다중 로봇시스템의 충돌 회피 문제에 적용하였다.

본 논문에서는 이동로봇이 주행할 경로의 궤적을 3차원 Bezier 곡선에 의해서 생성하는 문제를 다룬다. 시작점과 목표점의 위치와 자세가 주어지면 3차 Bezier곡선의 함수에 의해서 곡선을 생성한다. 시작점에서 목표점까지는 여러 곡률 값에 의해서 다양한 경로가 만들어 질 수 있다. 시작점과 목표점 사이에 놓이는 제어점들이 곡률을 결정하는 역할을 한다. 본 논문에서는 이 제어점들의 위치를 구하기 위한 방법이 제안된다. 이 점

들에 의해 생성된 경로점들은 기구학적인 제한의 범위 내의 값들이어야 하므로 제약조건에 대해서 검사를 한다.

2. 본 론

2.1 이동로봇의 기구학

두 개의 모터에 의해서 구동되는 로봇의 기구학은 다음과 같이 표현된다.

$$\dot{x}_c = v_c \cos(\theta_c) \tag{1}$$

$$\dot{y}_c = v_c \sin(\theta_c) \tag{2}$$

$$\dot{\theta} = w_c \tag{3}$$

여기서, v_c 는 로봇의 접선속도(tangential velocity)를 나타내며, w_c 는 각속도(angular velocity)를 나타낸다 [10]. 좌표(x_c, y_c)는 실제계 시스템에 대한 로봇의 위치 나타내고, θ_c 는 로봇의 조향각을 나타낸다. 이때 이것은 다음과 같이 나타낼 수 있다.

$w_c \neq 0$ 인 경우

$$x_c^{k+1} = x_c^k + \frac{v_c}{w_c} [\sin(\theta_c^{k+1}) - \sin(\theta_c^k)] \tag{4}$$

$$y_c^{k+1} = y_c^k - \frac{v_c}{w_c} [\cos(\theta_c^{k+1}) - \cos(\theta_c^k)] \tag{5}$$

$$\theta_c^{k+1} = \theta_c^k + w_c t_s \tag{6}$$

$w_c = 0$ 인 경우

$$x_c^{k+1} = x_c^k + v_c t_s \cos(\theta_c^k) \tag{7}$$

$$y_c^{k+1} = y_c^k + v_c t_s \sin(\theta_c^k) \tag{8}$$

$$\theta_c^{k+1} = \theta_c^k \tag{9}$$

여기서, k 는 샘플링 지수를 나타내고 t_s 는 샘플링 시간을 나타낸다. 로봇의 각 구동 휠이 최대 각 가속도 \dot{w}_{max} 에 의해 제한될 때, 접선 가속도 a_{max} 와 각 가속도 α_{max} 의 최대값은 다음과 같은 한계값을 갖는다.

$$a_{max} = \frac{r_w \dot{w}_{max}}{2} \tag{10}$$

$$\alpha_{max} = \frac{r_w \dot{w}_{max}}{d_w} \tag{11}$$

2.2 Bezier 곡선

차수가 n 인 일반적인 Bezier곡선의 다항식은 다음과 같이 정의된다.

$$P(u) = \sum_{i=0}^n B_{i,n}(u) p_i; \quad 0 \leq u \leq 1 \tag{12}$$

위의 식에서 각 p_i 들은 2차원 평면상에 주어진 점의 좌

표를 나타내고 기하학적 계수 b_i 를 조정점 혹은 제어점이라고 부른다. 기저함수 $B_{i,n}(u)$ 는 고전적인 n 차 Bernstein 다항식으로서 그 다항식은 다음과 같이 정의된다.

$$B_{i,n}(u) = C_{n,i} u^i (1-u)^{n-i} \quad (13)$$

여기서,

$$C_{n,i}(u) = \frac{n!}{i!(n-i)!} \quad (14)$$

이다. $n=3$ 인 경우에 위의 식들은 다음과 같이 표현된다.

$$\begin{aligned} B_{0,3} &= (1-u)^3 \\ B_{1,3} &= 3u(1-u)^2 \\ B_{2,3} &= 3u^2(1-u) \\ B_{3,3} &= u^3 \end{aligned} \quad (15)$$

위의 값들을 식(12)에 대입하면 다음과 같이 된다.

$$P(u) = (1-u)^3 p_0 + 3u(1-u)^2 p_1 + 3u^2(1-u) p_2 + u^3 p_3 \quad (16)$$

여기서, p_0 는 곡선의 시작점을 의미하고, p_3 은 끝점을 의미한다. 그 이외의 p_1, p_2 점 들은 곡선의 모양 형성에 주요한 역할을 하는 제어점이 된다.

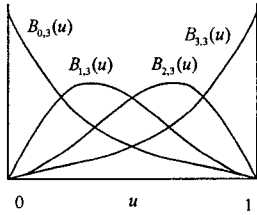


그림 1. 3차 Bezier 곡선에서의 배합함수

2.3 경로 생성을 위한 제어점의 결정

3차 Bezier 곡선은 4개의 점으로 구성된다. 이 점들을 각각 P_0, P_1, P_2, P_3 라고 하자. 여기서 시작점을 P_0 라고 하고, 목표점을 P_3 이라고 하자. 시작점과 목표점의 위치가 주어지고, 로봇이 향하고 있는 초기 자세 θ_0 와 목표 자세 θ_3 가 정해지면, 곡선경로를 생성을 위해서 두 개의 제어점이 결정되어야 한다(그림 2). 시작점 P_0 와 연관된 제어점 P_1 은 시작점의 접선 상에 놓이게 된다. 마찬가지로 목표점 P_3 와 연관된 제어점 P_2 는 목표점의 접선 상에 놓이게 된다. 제어점들의 방향은 로봇의 초기 자세와 목표 자세에 의해서 결정되므로 선분의 길이를 구하면 제어점의 위치가 결정될 수 있다.

시작점 P_0 의 좌표가 (x_0, y_0) 이고, 목표점 P_3 의 좌표가 (x_3, y_3) 일 때, 두점간의 거리를 m 등분한 값은 다음과 같이 결정된다.

$$l = \frac{\sqrt{(x_3 - x_0)^2 + (y_3 - y_0)^2}}{m} \quad (17)$$

여기서, m 은 시작점 P_0 와 끝점 P_3 간의 거리를 임의의 간격으로 등분하기 위한 값이다.

l 은 P_0 와 P_1 사이의 거리가 되고, 또한, P_2 와 P_3 사이의 거리가 된다. 수평선을 기준으로 P_0 와 P_1 이 이루는 각도는 θ_1 이고, P_2 와 P_3 가 이루는 각도는 θ_2 가 된다.

다. 그러므로 P_1 의 좌표는 P_0 을 기준으로 $(x_0 + l, y_0)$ 를 θ_1 만큼 반시계 방향으로 회전시킨 것이고, P_2 의 좌표는 P_3 를 기준으로 $(x_3 - l, y_3)$ 를 $180 - \theta_2$ 만큼 반시계 방향으로 회전시킨 것이다(식 (18)과 식 (19)).

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 \\ \sin \theta_1 & \cos \theta_1 \end{bmatrix} \begin{bmatrix} x_0 + l \\ y_0 \end{bmatrix} \quad (18)$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 \\ \sin \theta_2 & \cos \theta_2 \end{bmatrix} \begin{bmatrix} x_3 - l \\ y_3 \end{bmatrix} \quad (19)$$

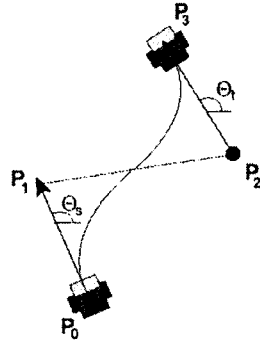


그림 2. 제어점의 결정

v_R 과 v_L 을 각각 우측바퀴와 좌측바퀴의 속도라고 하자. b 를 두 바퀴간의 거리라고 할 때, 로봇의 중심에서의 속도는 다음과 같이 된다.

$$v = \frac{v_R + v_L}{2} \quad (20)$$

이동로봇의 경로에 대한 제어를 하기 위해서는 두 바퀴의 속도를 제어하는 것과 이동한 거리를 측정하는 것이 필요하다. 속도와 거리는 이동로봇의 기구학적인 제약, 즉 최대 속도(v_{max}), 최대 가속도(a_{max}), 최대 허용 원심력(f_{max})에 의해서 영향을 받는다. 무게가 m 인 로봇의 속도 v 는 다음의 조건들을 만족해야 한다.

$$\begin{aligned} v &> 0 \\ |v_R| &< v_{max}, \quad |v_L| < v_{max} \\ |a_R| &< a_{max}, \quad |a_L| < a_{max} \\ m \frac{v^2}{f} &< f_{max} \end{aligned} \quad (21)$$

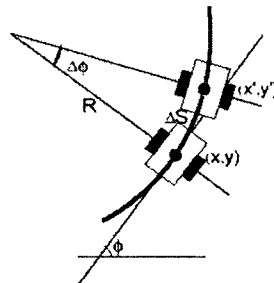


그림 3. Bezier 곡선의 어떤 점(x,y)와 다음 점(x',y')과의 관계

그림 3에서 어느 한점 $p(x,y)$ 에서 인접한 점 $p'(x,y)$ 과의 관계는 다음과 같이 나타낼 수 있다.

$$\begin{aligned} \Delta x &= R \sin \Delta \phi \\ \Delta y &= R(1 - \cos \Delta \phi) \\ x' &= x + \Delta x \cos \phi - \Delta y \sin \phi \\ y' &= y + \Delta x \sin \phi + \Delta y \cos \phi \end{aligned} \quad (22)$$

$u(0 \sim 1$ 사이의 값)의 값에 따라 두점간의 간격의 크기가 정해지므로 i 번째에서 식(22)에 의해 구한 두 점의 간격은 v_{max} 에 의한 거리값 이하의 간격을 가져야 한다.

$$|P_i(x, y) - P_{i+1}(x', y')| < S v_{max} \quad (23)$$

2.5. 시뮬레이션

다음의 시뮬레이션의 결과는 시작점에서 임의의 자세를 유지하고 있는 로봇이 목표지점에 지정된 각으로 도달할 수 있음을 입증하기 위한 예들이다. 이 시뮬레이션에서 $m=3$ 으로 사용하고 있다. 로봇의 최대속도는 $v_{max}=0.5[m/s]$ 이다.

그림 4의 A의 경우는 시작점이 원점에 놓여있고 $\theta_s=0.5\pi[rad]$ 인 상태에서 목표점이 $(1.2, 1.5)[m]$ 에 놓여있고 $\theta_t=0.5\pi[rad]$ 인 경우이다. 그림 4의 B는 시작점은 원점에 있고 $\theta_s=0[rad]$ 이다. 목표점은 $(1.2, 1.5)[m]$ 에 놓여있고, $\theta_t=0.5\pi[rad]$ 인 경우를 나타낸 것이다. 그림 4의 C는 시작점은 원점에 있고 $\theta_s=\pi[rad]$ 이고 목표점은 $(1.2, 1.5)[m]$ 에 놓여있고, $\theta_t=0[rad]$ 인 경우를 나타낸 것이다. 세 가지 경우 모두 제약조건을 만족하고 목표지점에 도달하였다.

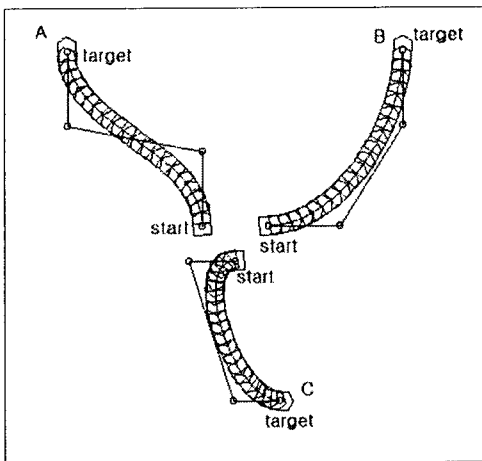


그림 4. 시작점과 끝점의 좌표와 자세에 따른 경로

그림 6은 제어점의 선택에 사용되는 m 값을 각각 2, 3, 5의 값을 선택한 경우를 비교 한 것이다. m 값이 크면 작은 곡률을 갖는 경로가 생성된다.

3. 결 론

본 논문에서는 이륜형 로봇이 임의의 경로를 따라 목표지점에 정해진 자세로 도달할 수 있는 경로를 생성하기 위하여 3차 Bezier 곡선을 이용하는 방법을 사용하였다. 이 곡선을 형성하는 주요한 요소인 제어점들에 의

해서 다양한 형태의 곡률을 갖는 경로가 생성된다. 이런 제어점의 위치를 결정하는 방법을 제안하고, 결과로 생성된 점들이 로봇의 이동을 위한 제약조건을 조사하였다. 시뮬레이션 결과를 통해서 본 논문의 방법으로 형성된 경로를 이륜형 로봇이 추종가능함을 입증하였다.

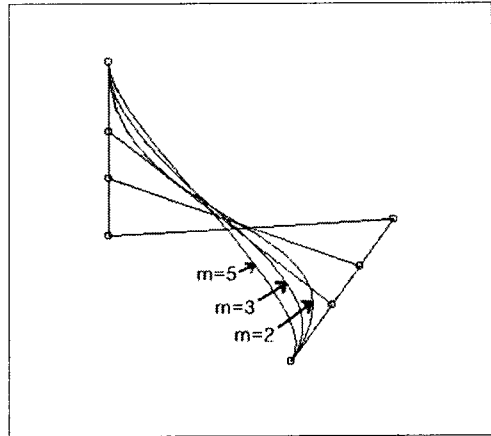


그림 5. 제어점의 선택에 사용되는 m 값의 비교

(참 고 문 헌)

- [1] A. Segovia, et.al, "Comparative study of the different methods of path generation for a mobile robot in a free environment", 5th International Conference on Advanced Robotics, 1991. 'Robots in Unstructured Environments', Vol. 2, pp. 1667 -1670, 1991
- [2] W. L. Nelson, "Continuous curvature paths for autonomous vehicles", Proc. of the IEEE Int. Conf. on Robotics and Automation, vol. 3, pp. 1260-1264, Scottsdale, AZ (US), May 1989.
- [3] Y. Kanayama and B.I. Harman, "Smooth local path planning for autonomous vehicles", Proc. of the IEEE Int. Conf. on Robotics and Automation, vol. 3, pp. 1265-1270, Scottsdale, AZ (US), May 1989.
- [4] Chun-Sin Lin et.al., "Formulation and optimization of cubic polynomial joint trajectories for industrial robots", IEEE Trans. on Automatic Control, Vol. 28, Dec 1983.
- [5] Arakawa, A. et.al., "Trajectory generation for wheeled mobile robot based on landmarks", IEEE International Conference on Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., vol. 2 .pp. 1273 -1278.
- [6] K.Komoriya and K. Tanie, "Trajectory Design and Control of a Wheel-type Mobile Robot Using B-spline Curve", IEEE/RSJ International Workshop on Intelligent Robots and Systems '89, Sep. 4-5, Tsukuba, Japan, pp. 398-405, 1989.
- [7] Eric Paulos, "On-line Collision Avoidance for Multiple Robots Using B-Spline", U.C. Berkeley Report, CSD-98-977, pp. 1-15, Jan. 1998.