Journal of Institute of Control, Robotics and Systems (2013) 19(6):540-546
http://dx.doi.org/10.5302/J.ICROS.2013.13.1860 ISSN:1976-5622 eISSN:2233-4335

물리적 제한을 고려한 두 바퀴 로봇의 관절 공간 궤적 생성 방법

Joint Space Trajectory Planning Considering Physical Limits for Two-wheeled Mobile Robots

양 길 진, 최 병 욱*
(Gil-Jin Yang[1] and Byoung-Wook Choi[2])
[1]Graduate School in Dept. of Electrical Engineering, Seoul National University of Science and Technology
[2]Dept. of Electrical and Information Engineering, Seoul National University of Science and Technology

Abstract: This paper presents a trajectory planning algorithm for TMR (Two-wheeled Mobile Robots). The trajectory is developed in joint space and considers the physical limits of a TMR. First, we present a process for generating a smooth curve through a Bezier curve. The trajectory for the center of the TMR following the Bezier curve is developed through a convolution operator taking into consideration its physical limits. The trajectory along the Bezier curve is regenerated using time-dependent parameters which correspond to the distance driven by the velocity of the center of the TMR in a sampling time. The velocity commands in the Cartesian space are converted to actuator commands for two wheels. In case that the actuator commands exceed the maximum velocity, the trajectory is redeveloped with compensated center velocity. We also suggest a smooth trajectory planning algorithm in joint space for the two segmented paths. Finally, the effectiveness of the algorithm is shown through numerical examples and application to a simulator.

Keywords: trajectory planning, Bezier curve, convolution operator, configuration space, mobile robots

I. 서론

최근 두 바퀴를 이용한 차동 구동 방식의 이동 로봇(TMR: Two-Wheeled Mobile Robot)이 로봇 청소기나 지능형 서비스 로봇 등에 활용이 많아지고 있다. 이를 위한 주행 시스템에 대한 연구가 많이 진행되고 있다.

TMR의 주행시스템은 크게 경로를 생성하는 경로계획기와 경로를 시간의 함수로 나타내는 궤적 생성기와 생성된 궤적을 추종하는 추종 제어기 그리고 구동 제어기로 나누어진다. 경로계획은 목표 작업 공간에서 원하는 자세를 유지하면서 부드러운 경로를 생성하는 문제이며, 궤적 생성기는 계획된 경로에 대하여 시간의 함수로 속도 프로파일을 생성하는 것이 목표이다. 추종 제어기와 구동 제어기는 이동로봇의 물리적 제한을 만족하면서 원하는 시간에 주어진 궤적으로 주행할 수 있도록 제어하는 시스템이다.

본 논문은 TMR의 주행 시스템에서 부드러운 곡선경로를 추종하는 궤적 생성 방법을 연구한다. TMR의 경로계획에 관한 문제는 작업공간 안에서 원하는 목표점으로 부드러운 주행을 하는 문제, 로봇의 물리적 제한을 만족하는 경로 계획 문제, 그리고 최단의 시간에 최소의 에너지를 이용하는 경로계획에 관한 연구가 활발히 진행되고 있다[1-10].

최근 궤적 생성에서 로봇의 물리적 제한을 고려함으로 로봇의 손상을 줄이며, 궤적 추종 정확도와 추종 속도를 향상시키기 위하여 콘볼루션 연산자를 이용하여 작업 공간에서 로봇의 물리적 제한을 고려한 속도 궤적 생성방법이 제안되었다[2-4]. 그러나 기존의 콘볼루션을 이용한 이동로봇의 궤적 생성 방법에서는 직교 좌표계에서 로봇 중심에 대하여 직선경로와 직선 속도만을 고려함으로써 TMR 로봇의 기구학 중 하나인 이동 로봇의 방향각이 고려되지 않았다. 초기 방향각 및 최종 목표점에서의 방향각을 고려한 부드러운 경로계획방법은 TMR의 경로계획의 기본적인 목표가 된다. 또한 직교 좌표계에서 이동로봇의 중심에 대한 직선 속도 제한만을 고려하여 회전 각속도의 변화에 따른 실제 로봇의 구동축인 두 바퀴에 대한 물리적 제한이 고려되지 않아서 실제 TMR로의 적용에 어려운 문제를 가지고 있다.

기존 TMR의 경로 계획은 직선과 원호 또는 클로소이드를 이용하여 경로를 생성하였다[8]. 이러한 방법은 곡선주행은 가능하지만 직선과 곡선의 연결부분에서 불연속점이 발생할 수 있으며, TMR의 궤적을 생성 시 불연속점이 발생하거나 이것으로 인한 미끄러짐, 편차 문제 등이 발생할 수 있다. 이를 극복하기 위하여 TMR의 연속 곡률을 갖는 경로에 대한 연구가 수행되었으며, 베지어 곡선(Bezier curve)을 이용하여 시작점에서의 자세와 목표점에서 원하는 자세로 주행 하는 경로계획방법이 연구되었다[9]. 그러나 기존의 경로계획방법에서는 일정한 값으로 증가하는 매개변수를 통해 경로를 생성하기 때문에 미리 정의된 중심속

* 책임저자(Corresponding Author)
Manuscript received January 4, 2013 / revised March 20, 2013 / accepted March 28, 2013
양길진: 서울과학기술대학교 전기공학과(yang6495@gmail.com)
최병욱: 서울과학기술대학교 전기정보공학과(bwchoi@seoultech.ac.kr)
※ 이 논문은 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단 기초연구사업의 지원을 받아 수행된 연구임(2012-006057).

도 궤적을 이용하여 경로를 추종하기에는 어려움이 있다.

본 논문에서는 TMR의 정확하고 부드러운 주행과 체계적인 경로 추종을 위하여 베지어 곡선을 이용한 경로 생성 방법을 이용한다. 베지어 곡선에 의하여 생성된 곡선경로에 대하여 물리적 제한을 만족하면서 주행하기 위하여 이동거리만을 고려한 중심속도 궤적을 콘볼루션 연산자를 이용하여 구한다. 그리고 생성된 중심속도 궤적으로 이동할 때의 부드러운 곡선경로의 회전각을 반영하기 위하여 베지어 곡선의 매개변수를 시간의 함수로 변환하는 방법을 제안한다. 이를 이용하여 베지어 곡선을 다시 생성하면 물리적 제한을 만족하면서 부드러운 곡선인 베지어 곡선을 추종하는 로봇의 이동 궤적을 생성할 수 있게 된다. 마지막으로 로봇의 중심속도를 회전 각속도에 따른 구동부의 물리적 제한을 고려한 두 바퀴 축의 구동 속도궤적을 생성하는 방법을 제시함으로써 실제 TMR 구동에 사용할 수 있는 궤적 생성 방법을 제안한다.

제안된 속도 궤적은 로봇의 물리적 제한을 만족하면서 회전 방향각에 따른 경로를 추종할 수 있으며, 또한 구동부의 물리적 제한을 만족하는 두 축의 궤적으로 변환이 가능하여 TMR 구동에 직접적으로 활용할 수 있다. 제안된 방법의 유효성을 위하여 수치적 시뮬레이션을 수행하였으며, 계획된 궤적을 로봇 시뮬레이터에 적용하여 물리적 제한을 만족하면서 원하는 작업을 원활히 수행함으로 보였다[11].

본 논문의 구성은 II 장에서는 콘볼루션을 이용하여 물리적 제한을 만족하는 로봇의 중심속도를 간단히 재정리한다. III 장에서는 회전각을 고려하여 부드러운 곡선 주행을 위한 베지어 곡선을 활용한 경로 생성 방법을 설명한다. IV 장에서는 베지어 곡선을 추종하면서 로봇의 물리적 제한을 고려한 중심속도 생성 방법을 제안한다. V 장에서는 IV 장에서 생성된 부드러운 경로에 대하여 관절 공간 내에서 TMR의 시스템 사양을 고려하여 구동부 속도 명령을 생성하는 방법을 기술한다. VI 장에서 V 장에서 제안하는 방법의 유효성 및 적합성을 시뮬레이션 및 시뮬레이터 적용을 통하여 검증하였다. 그리고 마지막으로 결론을 맺는다.

II. 속도 제한을 고려한 콘볼루션 기반 궤적 생성 방법

TMR의 물리적 제한을 만족하면서 부드러운 중심속도 궤적을 생성하기 위하여 콘볼루션의 특성을 이용한 궤적 생성 방법이 제안되었다[2-4].

콘볼루션을 이용하기 위하여 다음 식과 같이 사각파형의 함수 $y_0(t)$를 정의한다.

$$y_0(t)=\begin{cases}v_0, & 0\le t\le t_0\\ 0, & otherwise\end{cases} \tag{1}$$

여기에서 n번째 적용되는 콘볼루션 함수는 $h_n(t)$는 다음 식과 같이 $0\le t\le t_n$의 구간에서 단위 면적(unit area)을 갖는 사각파형 함수로 정의한다.

$$h_n(t)=\begin{cases}1/t_n, & 0\le t\le t_n\\ 0, & otherwise\end{cases} \tag{2}$$

여기에서 $t_n=\dfrac{v_{\max}^{(n-1)}}{v_{\max}^{(n)}}$으로 주어진다. 따라서 함수 $y_n(t)$를 단위 면적을 갖는 함수 $h_n(t)$와 n번 콘볼루션 했을 때 결과라면 속도함수의 최댓값은 $v_0=v_{\max}^{(0)}$과 같다. 그리고 $v_{\max}^{(1)}$는 1차 속도함수의 최댓값으로 $a_{\max}$ 최대 가속도를 의미하며 $v_{\max}^{(2)}$는 2차 속도함수의 최댓값인 $j_{\max}$ 최대 저크를 의미한다. 따라서 물리적 제한이 고려된 미분 가능한 S 곡선(S-curve)의 중심속도 궤적이 생성된다.

다음 식 (3)과 같은 디지털 콘볼루션 식을 이용하여 TMR의 중심속도를 생성한다[2].

$$\begin{aligned}y_n[k]&=\frac{y_{n-1}[k]-y_{n-1}[k-m_n]}{m_n}+y_n[k-1]\\ k&=[t/T_s]\qquad m_n=[t_n/T_s]\end{aligned} \tag{3}$$

여기에서 T_s는 샘플링 시간, [x]는 가우스 양자화 기호로 x 보다 크지 않은 최대 정수를 나타낸다.

III. 베지어 곡선을 활용한 로봇 방향 추적 궤적 생성

이동 로봇의 경로는 위치와 로봇의 방향각으로 구현된다. 따라서 이동 로봇의 경로는 시작점에서의 로봇의 위치와 방향각, 목표점에서의 로봇의 위치와 방향각이 고려되어야 하므로 3차 베지어 곡선을 이용하여 곡선 궤적을 생성한다[9].

그림 1과 같이 시작점 $P_i(x_i,y_i,\theta_i)$과 목표점 좌표 $P_f(x_f,y_f,\theta_f)$, 제어점 $C_1(x_1,y_1)$와 $C_2(x_2,y_2)$로 이루어진 3차 베지어 곡선을 이용하여 궤적을 생성한다. 다음은 베지어 곡선의 제어점을 결정하는 방정식이다.

$$\begin{aligned}x_1&=x_i+d_1\cos\theta_i\\ y_1&=y_i+d_1\sin\theta_i\\ x_2&=x_f+d_2\cos(180+\theta_f)\\ y_2&=y_f+d_2\sin(180+\theta_f)\end{aligned} \tag{4}$$

식 (4)에서 d_1, d_2는 각 시작점과 제어점, 목표점과 제어점 사이의 거리이며, d_1, d_2은 시작점과 목표점의 직선거리 L_d와 최대속도 $v_{\max}$와의 관계식으로 결정되며, 식 (5)와 같다.

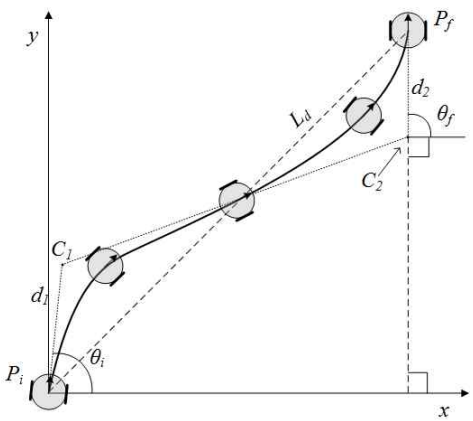

그림 1. 베지어 곡선.

Fig. 1. Bezier Curve.

$$d_1 = \left(1 - a_{\max}\left(1 - \frac{v_i}{v_{\max}}\right)\right) \times L_d/2$$
$$d_2 = \left(1 - a_{\max}\left(1 - \frac{v_f}{v_{\max}}\right)\right) \times L_d/2 \quad (5)$$

이렇게 결정된 제어점 C_1, C_2를 이용하여 베지어 곡선의 방정식을 계산한다. 베지어 곡선의 방정식은 다음 식 (6)와 같다.

$$\begin{aligned} x(u) &= \sum_{i=0}^{3} A_i J_{n,i}(u) \\ &= A_0(1-u)^3 + 3A_1 u(1-u)^2 + 3A_2 u^2(1-u) + A_3 u^3 \\ y(u) &= \sum_{i=0}^{3} B_i J_{n,i}(u) \\ &= B_0(1-u)^3 + 3B_1 u(1-u)^2 + 3B_2 u^2(1-u) + B_3 u^3 \end{aligned} \quad (6)$$

식 (6)에서 u는 $0 \le u \le 1$에서 임의의 값으로 일정하게 증가하는 값으로 설정하여 시작점에서 목표점까지의 부드러운 곡선을 생성할 수 있으며, 증가량이 적을수록 더욱 정밀한 베지어 곡선을 생성할 수 있다. 따라서 식 (6)으로 주어진 경로는 속도를 고려하지 않은 것이며 단지 u에 의하여 매개변수화 된 경로이다.

IV. 베지어 곡선과 중심속도 궤적의 합성 방법

회전각을 고려한 베지어 곡선으로 이루어진 경로에 대한 중심속도 궤적을 생성하기 위하여 식 (6)에 의하여 일정한 간격으로 u를 증가시켜 생성된 경로를 $\rho(u)$라고 정의한다. 그림 1의 P_i부터 P_f까지의 경로 $\rho(u)$를 추종하는 곡선 거리 B_d는 다음과 같이 계산된다.

$$\begin{aligned} B_d &= \sum_{u=0}^{1} \triangle\rho(u) \\ &= \sum_{u=0}^{1} \sqrt{(x(u+\triangle u) - x(u))^2 + (y(u+\triangle u) - y(u))^2} \end{aligned} \quad (7)$$

식 (7)에서 u값의 증가량이 적을수록 부드러운 곡선을 생성 한다. 계산된 B_d는 회전각을 고려한 실제 로봇의 이동 거리이다.

콘볼루션을 이용한 중심속도 궤적의 생성 방법에서 입력 값으로 중심 궤적의 이동거리를 S가 이용된다. 따라서 중심 궤적의 이동거리 $S = B_d$가 되도록 중심속도 궤적을 생성한다면 속도제한을 고려할 수 있는 콘볼루션의 장점을 이용하면서 부드러운 회전각을 유지하는 경로를 생성할 수 있게 된다. 여기서 v_i, v_f, $v_{\max}$, $a_{\max}$, $j_{\max}$, T_s는 임의로 설정이 가능하며, 사용하는 시스템의 사양을 이용하면 된다.

초기속도와 최종속도가 '0'이 아닌 경우 감가속 시간의 변화를 고려한 콘볼루션을 이용한 속도 생성 방법은 다음과 같다[4].

$$S_2 = (v_0 + v_i)t_0 + \frac{v_0}{2}(t_1 - t_1^+) + \frac{v_f + v_i}{2}(t_1 + t_2) \quad (8)$$

$$v_0 = sgn(S)v_{\max} - v_i \quad (9\text{-a})$$

$$t_2 = \frac{a_{\max}}{j_{\max}} \quad (9\text{-b})$$

$$t_1^* = \begin{cases} t_1^+, & 0 \le t \le t_0 \\ t_1, & t_0 \le t \le \sum_{k=0}^{n} t_k \end{cases} \quad (9\text{-c})$$

$$\begin{aligned} t_0 = \frac{sgn(S_2)}{v_{\max}} &\left(S_2 - \frac{v_f + v_i}{2}(t_1 + t_2) + \frac{v_i}{2}\left(t_1 - t_1^+\right)\right) \\ &- \frac{1}{2}(t_1 - t_1^+) \end{aligned} \quad (10)$$

식 (8)은 2번의 콘볼루션에 따른 이동 거리 S_2를 나타내며, 식 (9)-(10)은 콘볼루션 시 필요한 파라미터들을 나타낸다. 식 (9)에서 t_1^*은 가속도를 고려한 속도의 감가속 시간, t_2는 저크를 고려한 가속도의 감가속시간이다. 2번의 콘볼루션 결과 t_0에 t_1과 t_2의 시간이 더해짐으로 속도함수의 시간이 지연되지만 S_2는 콘볼루션의 특성에 의하여 주어진 목표 이동거리 S와 동일하게 이동하게 된다.

초기속도가 '0'이 아닌 경우 그림 2와 같이 S_2는 S_2^a와 S_2^b의 합과 같으며 식 (8)과 같이 나타낼 수 있다. v_0는 식 (9-a)와 같이 계산된다. 또한, 가속 시간 t_1^+ 및 감속 시간 t_1을 식 (11)과 같이 계산하여 조절함으로서 최대 가속도 활용이 가능하도록 한다. 주어진 최대 속도와 이동거리 S, 초기 속도 및 최종 속도를 갖고, 수식 (3)을 이용하여 TMR의 중심속도 함수 $v_c(t)$를 생성한다[4].

$$t_1^+ = \frac{v_{\max}}{a_{\max}} \quad (11\text{-a})$$

$$t_1 = \frac{v_{\max} - sgn(v_0)v_f}{a_{\max}} \quad (11\text{-b})$$

생성된 속도함수 $v_c(t)$는 방향을 고려하지 않고 이동 거리 S를 이동하기 위한 로봇의 중심속도 궤적이 된다. 방향을 고려하기 위하여 미소 경로 $\triangle\rho(t)$를 고려한다. 이는 중심속도에 따라 주어진 샘플링 기간 중 이동하여야 할 거리가 된다. 따라서 방향을 고려하기 위하여 속도 변화에 따른 이동경로에서의 베지어 함수의 매개변수 u를 결정하여야 하며, 이는 수식 (12)와 같이 구하여 진다. 여기서 결정된 $u(t)$를 베지어 곡선 수식 (6)에 입력함으로써 방향을 고려한 속도에 따른 궤적 $\rho(u(t))$를 구할 수 있다.

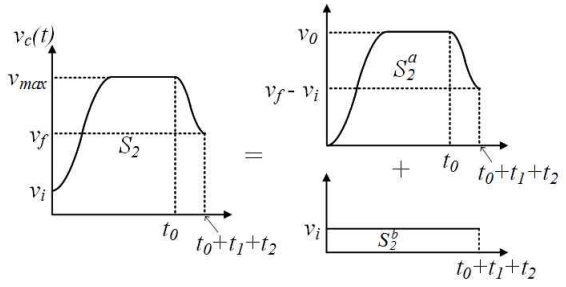

그림 2. 초기속도와 최종속도가 '0'이 아닌 콘볼루션.
Fig. 2. Convolution for non zero initial and final velocities.

$\rho(u(t))$는 샘플링주기가 적을수록 일정한 매개변수 값 u에 의하여 생성된 경로 $\rho(u)$를 더 잘 추종하게 된다.

$$u(t)=\frac{\sum_{t=0}^{t_0+t_1+t_2} v_c(t)}{B_d} \tag{12}$$

수식 (12)에서 $u(t)$는 $0 \le u(t) \le 1$ 구간에서 정의 되며 이동로봇 중심속도의 이동거리를 고려하여 매개변수의 증가량을 변환하는 것과 같은 의미이다. $u(t)$는 이미 물리적 제한을 만족하는 중심속도를 이용한 것으로 수식 (12)에 의하여 생성된 $\rho(u(t))$는 TMR의 물리적 제한을 만족하면서 부드러운 곡선을 추종하는 궤적이 된다.

V. 조인트 스페이스에서 두 바퀴 속도 명령 생성 방법

앞 장에서 물리적 제한을 만족하는 이동로봇의 중심속도를 이용하여 곡선주행이 가능한 궤적을 생성하였으며, 이를 이용하여 관절 공간에서 바퀴 속도 명령을 생성한다.

이동로봇의 위치는 전역 기준 좌표계와 로봇기준 좌표계로 구성되며 이동로봇의 위치 P_c는 전역 기준 좌표계 상에서 다음과 같이 정의 된다.

$$P_c=[x_c,\ y_c,\ \theta_c]^T \tag{13}$$

여기서 x_c, y_c, θ_c은 각각 이동로봇의 위치와 방향각을 나타내며, 차분 구동 형 이동로봇 두 바퀴의 기구학 모델은 다음과 같이 구할 수 있다.

$$P_c=\begin{bmatrix} \frac{r}{2}cos\theta_c & \frac{r}{2}cos\theta_c \\ \frac{r}{2}sin\theta_c & \frac{r}{2}sin\theta_c \\ \frac{r}{D} & -\frac{r}{D} \end{bmatrix}\begin{bmatrix} w_r \\ w_l \end{bmatrix} \tag{14}$$

여기서 r은 이동로봇 바퀴의 반지름을 의미하고, D는 이동로봇의 두 바퀴사이의 거리를 의미한다. w_r은 오른쪽 바퀴의 회전속도이고, w_l은 왼쪽 바퀴의 회전속도이다. 각 바퀴의 회전속도와 바퀴의 반지름을 이용하여 이동로봇의 각 구동 바퀴의 오른쪽 바퀴와 왼쪽 바퀴의 각속도를 식 (15)와 같이 계산할 수 있으며, 오른쪽 바퀴의 선속도 v_r, 왼쪽 바퀴의 선속도 v_l는 수식 (16)으로 주어진다.

$$\begin{aligned} w_r &= \frac{1}{r}(v_c + D/2 \cdot w_c) \\ w_l &= \frac{1}{r}(v_c - D/2 \cdot w_c) \end{aligned} \tag{15}$$

$$\begin{aligned} v_r &= rw_r \\ v_l &= rw_l \end{aligned} \tag{16}$$

식 (16)을 통해 계산된 두 바퀴의 선속도의 차이로 인하여 이동로봇의 각속도 w_c가 발생하며 다음 식 (17)과 같이 계산된다.

$$w_c=\frac{v_r-v_l}{D} \tag{17}$$

바퀴의 속도 명령 생성을 위한 베지어 곡선 궤적의 각 구간의 각도 $\triangle\theta$와 회전각속도 w_c는 다음 식 (18)과 같이 계산되며, w_c는 $\triangle\theta$를 샘플 시간에 따라 미분하여 계산할 수 있다.

$$\triangle\theta=\tan^{-1}\frac{y(\triangle u)}{x(\triangle u)} \tag{18-a}$$

$$w_c=\frac{\triangle\theta}{\triangle t} \tag{18-b}$$

VI. 조인트 스페이스에서의 시스템 사양을 고려한 속도 명령 생성방법

1. 단일 경로에서의 속도 명령 생성 방법

그림 4는 이동로봇의 물리적 제한이 $v_{\max}=0.5[m/s]$, $a_{\max}=0.2[m/s^2]$, $j_{\max}=0.2[m/s^3]$일 경우 시작점 (0.0, 0.0, 90°)에서 목표점 (4.0, 4.0, 90°)까지 이동하는 물리적 제한을 만족하는 로봇의 중심속도와 베지어 곡선 궤적을 따르는 두 바퀴의 속도 명령이다.

그림 4를 보면 베지어 곡선 궤적을 이용하여 생성된 속도 명령은 이동로봇 중심의 물리적 제한을 만족하지만 회전각이 있는 경우에 두 바퀴의 물리적 제한을 만족하지 못하므로 TMR의 구동축의 속도 명령으로 적합하지 않다.

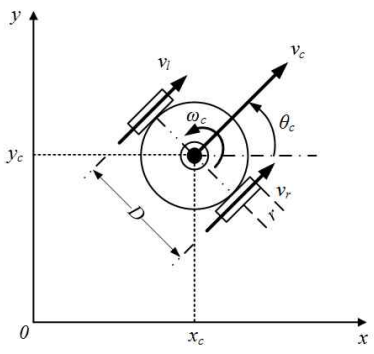

그림 3. TMR의 기준 좌표계.
Fig. 3. Reference frame of TMR.

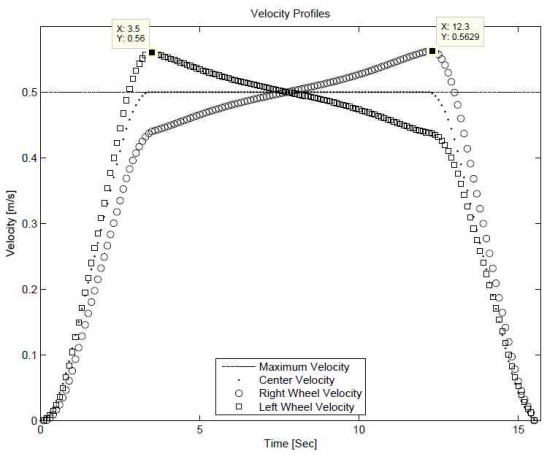

그림 4. 곡선 궤적을 따르는 관절 속도 명령.
Fig. 4. Joint velocity commands following curve trajectory.

따라서 관절 공간인 두 바퀴의 구동 속도 명령에 대하여 물리적 제한을 만족하는 속도 궤적이 필요하게 된다.

기존의 두 바퀴의 속도 명령 생성 시 곡선 주행으로 인하여 생기는 물리적 제한을 만족하지 못하는 값을 계산하여 중심속도 궤적 생성 시 다음 식 (19)와 같이 최대속도제한에서 보정속도 만큼 제한함으로서 물리적 제한을 만족하는 궤적을 생성 할 수 있다.

$$v_{comp} = \frac{|\max(v_r - v_l)|}{2} \tag{19-a}$$

$$v'_{\max} = v_{\max} - v_{comp} \tag{19-b}$$

중심속도 궤적 생성 시 최대 속도를 식 (19-b)와 같이 생성하여 곡선주행 시 물리적 제한을 만족하도록 하며, 이와 같은 방법으로 그림 4의 속도 명령에서 두 바퀴 모두 물리적 제한을 만족하도록 생성된 속도 명령은 그림 5와 같고, 생성된 속도 명령에 의한 이동로봇의 이동거리는 콘볼루션의 특성에 의하여 일정하다. 다만 주행 시간이 지연된다.

그림 6은 단일 경로에서 TMR의 이동궤적을 $x-y$좌표계로 나타낸 것이다. 베지어 커브로 계획된 부드러운 곡선주행을 하며, 시작점에서 목표점까지 원하는 방향각을 추종하면서 주행하는 것을 확인 할 수 있다.

2. 연속 경로에서의 속도 명령 생성 방법

이동로봇이 장애물이나 이동 불가능한 구간이 있을 경우 원하는 목표점까지 주행하기 위해서는 여러 경로를 추종할 수 있어야 한다. 이 때 이동로봇이 정지하지 않고 각 경로점을 연속적으로 지나가기 위하여 초기속도와 최종속도가 '0'이 아닌 경우를 이용하여 TMR의 두 바퀴의 속도 명령을 생성하는 방법을 제안한다.

그림 7은 시작점 (0.0, 0.0, 45°)에서 (2.0, 3.0, 30°)까지 이동 한 경로 A와 연속적으로 (2.0, 3.0, 30°)에서 (4.0, 4.0, 45°)으로 이동하는 경로 B를 베지어 곡선궤적 생성방법에 의해 회전각이 연속이 되도록 생성된 궤적의 각도와 각속도이다.

경로 A와 경로 B가 연속적으로 생성되지 않을 경우 각속도가 불연속적으로 생성이 되며, 두 바퀴의 속도가 불연속적으로 연결 되는 경우가 발생함으로 경로 생성에 유의하여야 한다. 그림 8은 경로 A와 경로 B가 연속적으로 생성되었을 경우 이다.

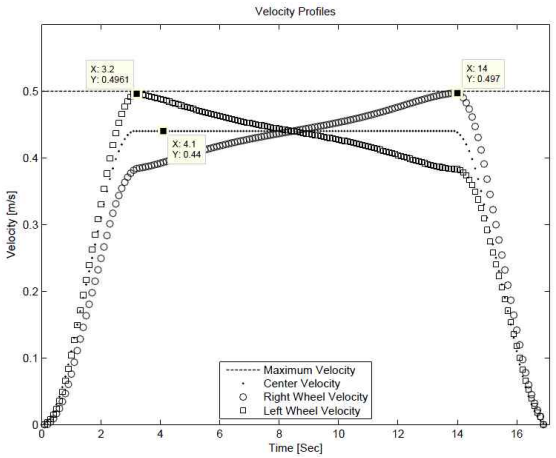

그림 5. 시스템 제한이 고려된 관절 속도 명령.

Fig. 5. Joint velocity commands considering system limits.

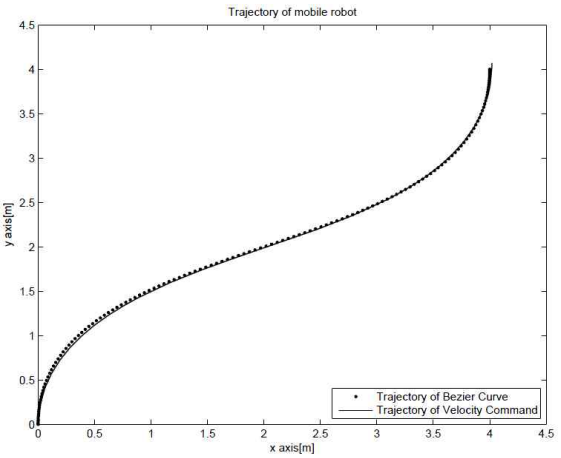

그림 6. 직교좌표계에서 TMR의 궤적.

Fig. 6. TMR trajectory in Cartesian space.

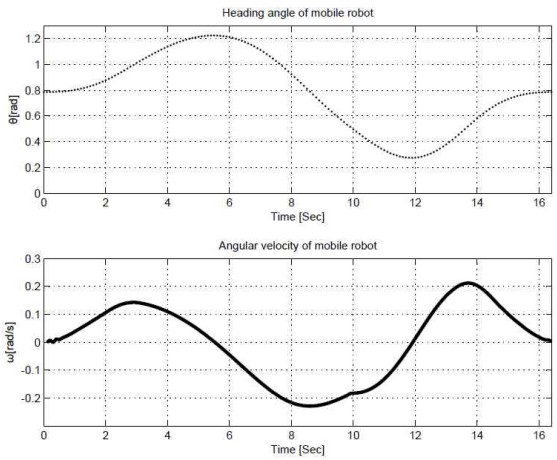

그림 7. 연속 경로에서 각도와 각속도.

Fig. 7. Heading angle and angular velocity in segmented path.

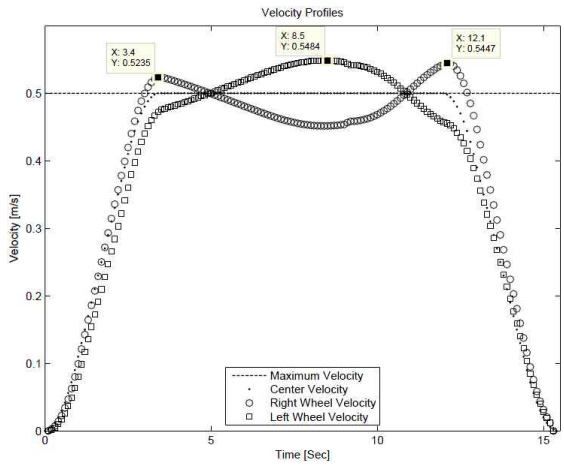

그림 8. 콘볼루션을 통하여 구한 관절 속도 명령.

Fig. 8. Joint velocity commands with convolution.

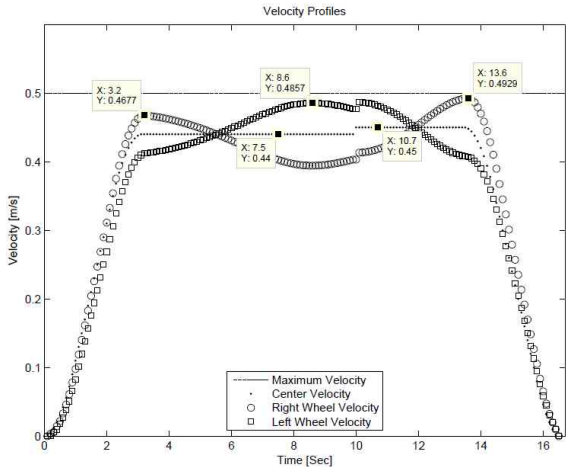

그림 9. 물리적 제한을 만족하는 관절 속도 명령.

Fig. 9. Joint velocity commands considering physical limits.

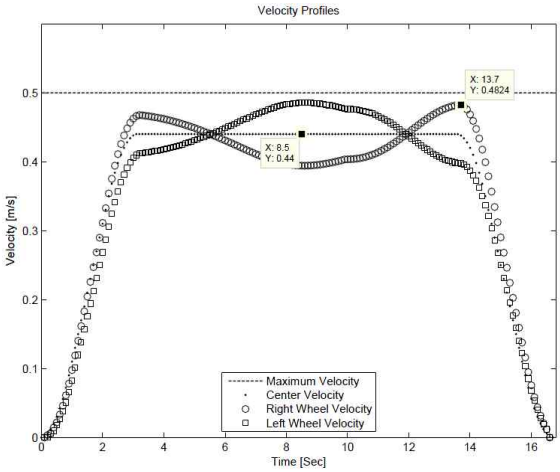

그림 10. 보정된 최대속도를 이용한 관절 속도 명령.

Fig. 10. Joint velocity commands with modified maximum velocity.

본 연구에서는 각 경로 A와 경로 B가 연속적으로 생성되었을 경우에 대하여 시스템 제한을 만족하는 이동로봇의 두 바퀴의 속도 명령 생성방법을 제안한다.

연속경로에서 연속적인 각도에 의하여 두 바퀴의 속도 명령이 생성되었을 경우 각 경로 A, B에서의 보정속도 v_{comp}^{A}와 v_{comp}^{B}라고 하며 각 구간에서의 물리적 제한을 만족하도록 최대 속도를 감소시켜 속도 명령을 그림 9와 같이 만들었다. 그러나 각 경로에서의 보정 속도의 차이에 의하여 구동축 속도 명령이 그림 9와 같이 불연속 속도 명령이 된다. 따라서 이와 같은 문제점을 해결하여야 구동축에서 속도 제한을 만족하는 속도 명령의 궤적을 생성할 수 있다.

본 연구에서는 물리적 제한을 만족하면서 부드럽게 경로점을 지나 최종 목표점 까지 이동하는 TMR을 위한 관절 속도 명령의 생성 방법을 제안한다.

다음 식 (20)과 같이 보정속도의 차이가 최소 속도 변화량을 만족하면 각 경로별 속도 보정 값을 그대로 사용하게 된다. 만일 최소 속도 변화량 이상일 경우는 보정 중심속도의 최대속도를 수식 (21)과 같이 결정한다. 식 (20)에서 $\triangle v_{\min}$은 이동로봇이 최대 저크에 의하여 최대 가속도 일 때 최대 속도 변화량이다.

$$\left| v_{comp}^{A} - v_{comp}^{B} \right| \le \left| \triangle v_{\min} \right| \tag{20}$$

$$v_{\max} = \begin{cases} v_{\max} - v_{comp}^{A} \text{ for } \left| v_{comp}^{A} \right| > \left| v_{comp}^{B} \right| \\ v_{\max} \quad\quad\quad\quad \text{for } \left| v_{comp}^{A} \right| = \left| v_{comp}^{B} \right| = 0 \\ v_{\max} - v_{comp}^{B} \text{ for } \left| v_{comp}^{A} \right| < \left| v_{comp}^{B} \right| \end{cases} \tag{21}$$

그림 10에서 구동축의 최대 속도 제한을 초과하여 새로운 구동축 속도 명령이 필요한 경우 수식 (20)에 의하여 보정량의 차이를 검증한다. 본 예제에서 최소 속도 변화량을 초과하여 수식 식 (21)을 이용하여 최대속도를 보정한다. 이를 이용하여 속도 명령을 생성하면 그림 12와 같이 연속적이며 제한을 만족하는 속도 명령을 생성할 수 있다.

그림 10으로 계획된 관절 공간 즉 구동축에 대한 속도 명령을 이용하여 실제 구동한 로봇의 이동 경로를 그림 11에 나타내었다. 최대 속도를 고려함에 따라 최초로 계획된

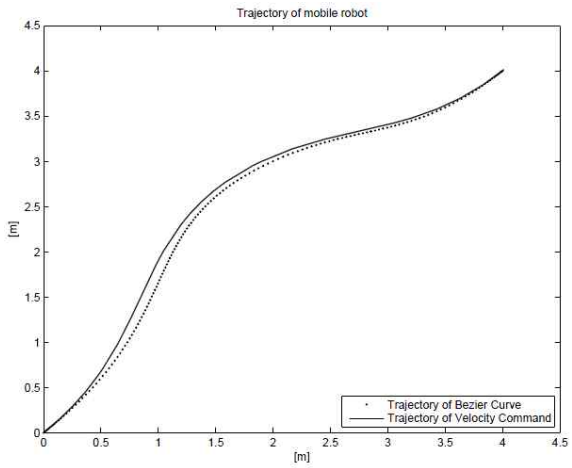

그림 11. 연속적인 경로에서 TMR의 궤적.

Fig. 11. Trajectory of TMR for continuous path.

경로와 약간의 오차를 보이고 있으나 오차 범위 내에서 로봇의 물리적 제한을 만족하면서 원하는 경로를 이동함을 알 수 있다. 실제 베지어 커브로 계획한 경로는 로봇의 물리적 제한으로 구동이 불가능한 경로이며 이를 관절 공간에서 구동 가능한 속도 명령의 궤적을 만들어서 구동하는 것은 적용 관점에서 매우 중요한 결과이다.

제안된 방법을 이용하여 이동로봇의 주행 시뮬레이션은 anyKode의 Marilou Robotics Studio를 이용하여 수행하였으며, 사용된 TMR은 $r = 10[cm]$, $D = 40[cm]$, $v_{\max} = 0.5[m/s]$, $a_{\max} = 0.2[m/s]$, $j_{\max} = 0.2[m/s]$로 설정하고, 경로점$(2, 3, 30^\circ)$에서 정지하지 않고 최종 목표점$(4, 4, 45^\circ)$까지 이동하는 시뮬레이션 결과 그림 12와 같이 회전각을 포함하여 원하는 경로를 추종하고 있음을 알 수 있다. 그림 12에서 각 격자 간격은 $0.5[m]$이며, 시뮬레이션 결과 이동로봇의 최종 목표점에서 오차는 x축 0.00892$[m]$, y축 0.01416$[m]$로 효율적으로 속도 명령이 생성되는 것을 확인할 수 있다.

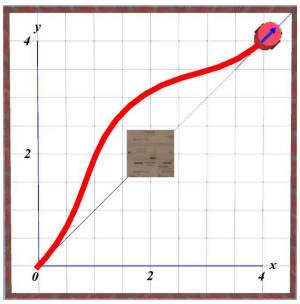

그림 12. 연속경로에서 TMR의 시뮬레이션 결과.
Fig. 12. Simulation result of TMR for continuous path.

VII. 결론

본 논문에서는 TMR을 위하여 관절 공간인 구동축의 속도 명령 궤적 생성 방법을 제안하였다. 궤적은 속도 및 가속도 그리고 저크의 시스템 제한을 만족하면서 부드러운 곡선 주행이 가능하며, 추후 동역학을 고려하면 실제 경로 제어에 적용이 가능한 유용한 결과이다.

제안된 관절 공간의 속도 궤적 생성 방법은 먼저 콘볼루션의 특성을 이용하여 로봇 중심의 속도에 대하여 최대 속도를 만족하도록 생성한다. 특히 이동로봇이 부드럽게 주행할 수 있도록 베지어 곡선를 추종하도록 로봇의 중심속도 궤적을 생성하였다. 그러나 로봇 중심속도는 작업 공간에서의 속도로서 실제 구동을 위하여 두 바퀴의 속도 궤적을 생성하여야 한다. 이와 같은 관절 공간의 속도 궤적은 회전각속도에 따라 두 바퀴의 속도가 변화되며, 이 때 구동 입력의 속도 제한을 만족할 필요가 있다. 따라서 관절 공간에서 실제 이동로봇이 추종할 수 있는 속도 명령을 생성하는 것이 중요하게 된다. 결과적으로 생성된 중심속도 궤적과 회전각을 이용하여 두 바퀴의 관절 공간에서의 속도 명령을 실제 물리적 제한을 만족하는 궤적을 생성하였다.

또한 실용성을 위하여 장애물이 있을 경우와 같은 연속적인 두 개의 곡선경로에 대하여 관절 공간에서 물리적 제한을 만족하는 속도 명령을 계획함으로써 실제 적용이 가능한 알고리즘임을 보였다.

향후 동역학을 고려하고 계획된 경로를 추종하는 제어 알고리즘을 구현하면 이동 시간 및 에너지 관점에서의 궤적생성 방법으로 확장이 가능할 것으로 기대한다.

REFERENCES

[1] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, 3rd Ed., Pearson Education, 2004.

[2] G. Lee, Y. Choi, and J. H. Kim, "Convolution-based desired trajectory generation method considering system specifications," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 16, no. 10, pp. 997-1005, Oct. 2010.

[3] G. Lee, D. Kim, and Y. Choi, "Faster and smoother trajectory generation considering physical system limits under discontinuously assigned target angles," *IEEE International Conference on Mechatronics and Automation*, pp. 1196-1201, Aug 2012.

[4] G. Lee, "Development of trajectory generation method using digital convolution and its robotic application to obstacle avoidance," A Master's Thesis of Hanyang University, Aug. 2010.

[5] M. Brezak and I. Petrovic, "Time-optimal trajectory planning along predefined path for mobile robots with velocity and acceleration constraints," *IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, Budapest, Hungary, pp. 942-947, 2011.

[6] M. Lepetic, G. Klancar, I. Skrjanc, D. Matko, and B. Ptocnik, "Time optimal path planning considering acceleration limits," *Robotics and Automation Systems*, vol. 45, pp. 192-210, 2003.

[7] J. S. Kim and B. K. Kim, "Efficient minimum-time cornering motion planning for differential-driven wheeled mobile robots with motor control input constraint," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 19, no. 1, pp. 56-64, Jan. 2013.

[8] J. S. Kim and B. K. Kim, "Minimum-time grid coverage trajectory planning algorithm for mobile robots with battery voltage constraints," *International Conference on Control,Automation and System*, pp. 27-30, Oct. 2010.

[9] K. G. Jolly, R. Sreerama Kumar, and R. Vijayakumar, "A bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits," *Robotics and Automation Systems*, vol. 57, pp. 23-33, Jan. 2009.

[10] K. Lee, D. Kim, and K. H. Rew, "Trajectory planning of multi agent robots for robot soccer using complex potential," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 12, pp. 1073-1078, Dec. 2012.

[11] anyKode, Marilou Robotics Studio www.anykode.com

양 길 진

2013년 원광대학교 전자공학과 졸업. 2013년~현재 서울과학기술대학교 전기공학과 석사과정. 관심분야는 실시간 시스템 설계, 임베디드 리눅스, 지능형 로봇 소프트웨어.

최 병 욱

1988년~1992년 한국과학기술원 전기및 전자공학과 석사 및 박사졸업. 1988년~2000년 LG산전 중앙연구소 책임연구원. 2000년~2005년 선문대학교 제어계측공학과 부교수. 2003년~2005년 임베디드웹 대표이사. 2007년~2008년 Nanyang Technological University, Senior Fellow. 2005년~현재 서울과학기술대학교 전기정보공학과 교수. 관심분야는 실시간 시스템 설계, 임베디드 시스템, 임베디드 리눅스, 지능형 로봇 소프트웨어.