JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지(J. Korea Inst. Inf. Commun. Eng.) Vol. 21, No. 4 : 698~707 Apr. 2017

유전알고리즘을 이용한 이동로봇의 예측제어

손현식[1] · 박진현[2] · 최영규[3*]

Predictive Control for Mobile Robots Using Genetic Algorithms

Hyun-sik Son[1] · Jin-hyun Park[2] · Young-kiu Choi[3*]

[1]Department of Electrical and Computer Engineering, Pusan National University, Busan 46241, Korea
[2]Dep. of Mechatronics Engineering, Kyeongnam National University of Science and Technology, Jinju 52725, Korea
[3*]Department of Electrical Engineering, Pusan National University, Busan 46241, Korea

요 약

본 논문에서 이동로봇의 기준궤적추적제어를 위한 예측제어방법을 다룬다. 예측제어는 예측모델을 사용하여 기준궤적과 시스템 상태 간의 미래오차들을 최소화시키는 효과적인 제어방법으로 알려져 있으나, 실시간 계산량이 너무 많아 화공정 플랜트와 같이 매우 느린 시스템에 한정되어 적용되었다. 근래에는 컴퓨터 기술 발달로 고속계산이 가능하여 이동로봇과 같은 빠른 시스템에도 예측제어방법이 도입되고 있다. 그런데 예측제어기에서 제어성능과 관계된 제어 파라미터들이 있는데 임의로 지정되어 최적화되지 못하였다. 본 논문에서 이동로봇 예측제어기 성능 개선을 위해 관련 제어 파라미터들을 유전알고리즘으로 최적화시켰고 모의실험을 통해 제어성능이 개선됨을 확인하였다.

ABSTRACT

This paper deals with predictive control methods of mobile robots for reference trajectory tracking control. Predictive control methods using predictive model are known as effective schemes that minimize the future errors between the reference trajectories and system states; however, the amount of real-time computation for the predictive control are huge so that their applications were limited to slow dynamic systems such as chemical processing plants. Lately with high computing power due to advanced computer technologies, the predictive control methods have been applied to fast systems such as mobile robots. These predictive controllers have some control parameters related to control performance. But these parameters have not been optimized. In this paper we employed the genetic algorithm to optimize the control parameters of the predictive controller for mobile robots. The improved performances of the proposed control method are demonstrated by the computer simulation studies.

키워드 : 이동로봇, 예측제어, 유전알고리즘, 제어파라미터

Key word : mobile robots, predictive control, genetic algorithm, control parameters

Received 29 November 2016, **Revised** 01 December 2016, **Accepted** 19 December 2016

* **Corresponding Author** Young-kiu Choi(E-mail:ykichoi@pusan.ac.kr, Tel:+82-51-510-1445)
Department of Electrical Engineering, Pusan National University, Busan 46241, Korea

Open Access https://doi.org/10.6109/jkiice.2017.21.4.698 print ISSN: 2234-4772 online ISSN: 2288-4165

ⓒThis is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(http://creativecommons.org/li-censes/ by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright Ⓒ The Korea Institute of Information and Communication Engineering.

Ⅰ. 서 론

로봇은 산업적 요구에 부응하여 연구되기 시작하였으며 주로 자동화를 위한 로봇 매니퓰레이터에 집중되어왔다. 근래에는 인간을 대신하기 위해 지능과 이동성을 갖춘 이동로봇에 대한 연구가 활성화되고 있다[1,2].

이동로봇의 연구 분야는 시각시스템 등의 각종센서 관련 응용 연구, 지능시스템 연구, 기구학과 동력학 관련 동적제어 연구 등이 있다[2]. 이동로봇의 동적제어 연구는 궤적 추종 제어(trajectory tracking control)과 자세 안정 제어(posture stabilizing control)로 대별될 수 있다[3]. 궤적 추종 제어는 미리 정해진 기준궤적 또는 이동하는 가상목표에 이동로봇의 위치와 방향이 수렴하도록 제어기를 구성한다. 반면에 자세 안정 제어는 기준목표점에 도달하게 하는 제어방법으로서 궤적 추종 제어보다 더 어려우며 이는 이동로봇의 nonholonomic 제약조건에 따라 어떤 순간에 특정방향 속도를 갖기 어렵기 때문이다. 이러한 이동로봇의 동적제어에 대해 폭넓은 연구[3-7]가 진행되어 왔으나 이동로봇의 상태와 입력의 제한조건에 대한 직접적 고려가 없어서 실제 적용시의 문제를 내포하고 있다. 예측제어는 제어시스템의 예측모델을 사용하여 기준궤적과 상태의 미래 오차까지 고려하여 현재 입력을 최적화시키면서 상태와 입력의 제한조건도 고려할 수 있는 매우 실용적인 방법이다. 한정된 예측구간에 대해서 미래오차를 최소화시키는 제어입력 수열을 구하고 그중에서 첫 번째 입력만을 사용하며 이러한 작업을 매 샘플링시간마다 반복한다. 이러한 예측제어는 제어성능이 뛰어나지만 실시간 계산량이 너무 많아서, 화학공정제어와 같이 시정수가 아주 많이 커서 출력응답이 매우 느린 시스템에 국한되어 적용되었다[8]. 그러나 근래에 컴퓨터 기술의 발달로 인해 매우 빠른 계산처리가 가능하여짐에 따라 예측제어방법이 로봇과 같이 빠른 시스템에도 적용가능하며 이에 대한 연구가 진행되고 있다[9-12]. 실시간으로도 이동로봇의 예측제어가 구현된 바 있으나 제어성능을 결정하는 여러 제어 파라미터들을 임의로 지정함에 따라 성능이 최적화되지 못하는 부분이 있었다[13].

본 논문에서 이동로봇의 궤적추종제어에 최적화 기능이 뛰어난 유전알고리즘[14]을 적용하여 추종오차를 최소화하고자 한다. 기준궤적은 두 번 미분 가능한 시간함수로 지정되며, 기준궤적 부근에서 선형화된 오차모델에 기초한 예측제어기가 사용된다. 이러한 예측제어기는 해석적으로 유도될 수 있으며 실시간 계산량이 비교적 많지 않기 때문에 실제 구현이 용이하다. 제어성능에 관계된 기준모델 파라미터 등을 유전알고리즘 염색체로 모델링하고 최적화하여 성능개선을 한다. 컴퓨터 시뮬레이션을 통해 본 논문에서 제안한 제어방법의 성능을 확인한다.

Ⅱ. 이동로봇의 모델링

2.1. 이동 로봇의 기구학

본 연구에서는 이동로봇의 예측제어 문제를 다루기 위해 그림 1과 같은 이동로봇의 모델을 고려한다. 먼저 예측제어기를 설계하기 위해 이동 로봇의 기구학을 수학적으로 나타낸다.

이동로봇의 자세는 식(1)과 같이 직교좌표와 각도로 나타낼 수 있다.

$$q = [x \quad y \quad \theta]^T \tag{1}$$

여기서 x와 y는 이동로봇의 무게중심위치, θ는 이동로봇의 진행방향에 대한 각도를 나타낸 것이다.

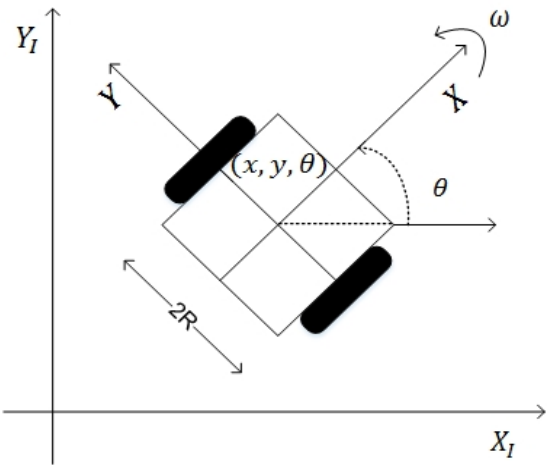


Fig. 1 Mobile Robot

식(2)와 같이 기구학 방정식으로 이동로봇의 속도와 자세의 관계를 나타낼 수 있다.

$$\dot{q} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix} \tag{2}$$

여기서 v와 ω 는 각각 이동로봇의 선속도와 각속도를 나타낸다.

2.2. 이동 로봇의 동역학

이동로봇의 동역학 특성에 의해 안정적인 성능을 보장할 수 없다. 따라서 안정적인 성능을 위해서 이동로봇의 동역학 특성을 고려해야한다.

본 논문에서는 로봇에 가해지는 외부의 힘과 마찰력, 미끄러짐 현상이 없다고 가정하였고 로봇이 수평면상에서 이동한다고 가정하였다. 식(3)과 같이 이동로봇의 동역학 식을 나타낼 수 있다.

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} = B(q)\tau - A^T(q)\lambda \tag{3}$$

여기서 $M(q)$는 관성행렬, $V_m(q,\dot{q})$는 원심력과 코리올리스 행렬(centrifugal force and Coriolis matrix)이다. 또한 $A^T(q)$는 제약에 관계된 행렬, $B(q)$는 입력에 대한 변환행렬, τ는 좌우바퀴에 가해지는 제어토크, λ는 구속력 벡터를 나타낸다[2].

식(4)와 같이 상태방정식을 유도하여 이동로봇의 위치와 각도를 구한다.

$$\begin{aligned} \dot{Z}_1 &= Z_2 \\ \dot{Z}_2 &= M^{-1}(q)\,(-V_m(q,\dot{q})\dot{q} - A^T(q)\lambda + B(q)\tau) \end{aligned} \tag{4}$$

단, $Z_1 = q = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}$, $Z_2 = \dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} z_4 \\ z_5 \\ z_6 \end{bmatrix}$

$\tau = \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix}$

비선형 미분방식인 식(4)를 풀어서 변수들을 구하고 이동로봇의 기구학 방정식인 식(2)를 이용하면 이동로봇의 선속도와 각속도를 식(5)과 같이 구할 수 있다.

$$\begin{aligned} v &= \dot{x}\cos\theta + \dot{y}\sin\theta \\ \omega &= \dot{\theta} \end{aligned} \tag{5}$$

그리고 식(6)에 있는 이동로봇의 선속도와 각속도를 이용하여 식(7)과 같이 이동로봇의 좌우바퀴의 각속도로 변환할 수 있다.

$$\begin{aligned} v &= \frac{v_r + v_l}{2} = \frac{r}{2}(\omega_r + \omega_l) \\ \omega &= \frac{v_r - v_l}{2R} = \frac{r}{2R}(\omega_r - \omega_l) \end{aligned} \tag{6}$$

$$\begin{aligned} \omega_r &= \frac{1}{r}(v + R\omega) \\ \omega_l &= \frac{1}{r}(v - R\omega) \end{aligned} \tag{7}$$

여기서 r은 이동로봇의 바퀴의 반지름, $2R$은 이동로봇의 바퀴 사이의 거리를 나타낸다. 또 ω_r과 ω_l은 이동로봇의 좌우 바퀴의 각속도를 나타낸다.

Ⅲ. 이동로봇 제어기설계

3.1. 피드포워드 제어

피드포워드 제어의 개루프 제어는 nonholonomic 시스템을 다루기 위한 직관적인 방법이다. 주어진 기준궤적 $[x_r(t), y_r(t)]$가 $t \in [0, T]$일 때 피드포워드 제어방법을 구할 수 있다. 식(2)의 기구학 모델을 이용하여 피드포워드 제어기 입력 $v_r(t)$과 $\omega_r(t)$을 구할 수 있다. 접선속도성분 $v_r(t)$는 식(8)과 같이 나타낼 수 있다.

$$v_r(t) = \pm\sqrt{\dot{x}_r^2(t) + \dot{y}_r^2(t)} \tag{8}$$

여기서 (+)는 정방향이고 (-)일 때는 역방향을 나타낸다. 그리고 경로의 각 지점에 대한 접선각은 식(9)와 같이 나타 낼 수 있다.

$$\theta_r(t) = \arctan2(\dot{y}_r(t), \dot{x}_r(t)) \tag{9}$$

여기서 arctan2는 4상한 역 탄젠트 함수이다. 식(9)을 미분하면 식(10)과 같이 로봇의 각속도 $\omega_r(t)$를 얻을 수 있다.

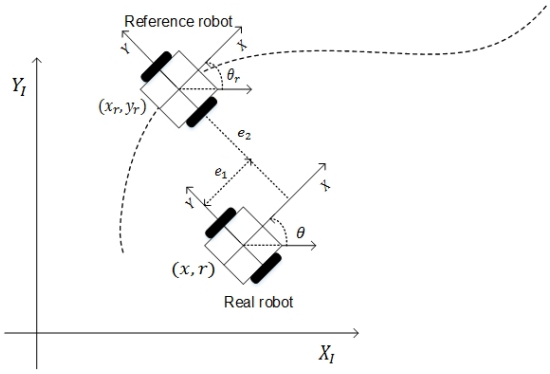


Fig. 2 Robot following error transformation

$$\omega_r(t) = \frac{\dot{x}_r(t)\ddot{y}_r(t) - \dot{y}_r(t)\ddot{x}_r(t)}{\dot{x}_r^2(t) + \dot{y}_r^2(t)} \tag{10}$$

경로 설계 과정에서 필요한 조건은 시간에 대해 두 번 미분 가능한 경로와 접선속도 $v_r(t)$가 0이 아니어야 한다[13].

3.2. 피드백 제어

그림 2와 같이 실제로봇의 좌표계에서 표현한 상태 추적오차 $e(t) = [e_1(t)\ e_2(t)\ e_3(t)]^T$는 식(11)과 같이 된다.

$$e = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}(q_r - q) \tag{11}$$

여기서 q_r은 기준로봇의 자세이고, q는 실제로봇의 자세이다.

그림 2에서 기준로봇은 기준경로 위를 이상적으로 움직이는 가상로봇이다. 반대로 실제로봇은 기준경로를 추적할 때 오차를 가진다. 그러므로 제어알고리즘은 실제로봇이 기준경로를 정확하게 따르도록 설계되어야 한다.

식(2)와 식(11)의 관계에서 식(12)을 구할 수 있다[13].

$$\dot{e} = \begin{bmatrix} \cos e_3 & 0 \\ \sin e_3 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} v_r \\ \omega_r \end{bmatrix} + \begin{bmatrix} -1 & e_2 \\ 0 & -e_1 \\ 0 & -1 \end{bmatrix} u \tag{12}$$

여기서 $u = [v\ \omega]^T$는 속도 입력 벡터이다. 로봇의 속도 입력 벡터 u는 식(13)과 같이 피드포워드와 피드백 제어동작의 합으로 구할 수 있다.

$$u = u_F + u_B \tag{13}$$

여기서 피드포워드 입력벡터 $u_F = [v_r\ \omega_r \cos e_3]^T$는 기준입력의 비선형변환으로 구할 수 있고, 피드백 입력벡터 $u_B = [u_{B_1}\ u_{B_2}]^T$로 구할 수 있다.

식(12)와 식(13)을 이용하여 식(14)와 같은 추적오차 모델을 구할 수 있다.

$$\dot{e} = \begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} e + \begin{bmatrix} 0 \\ \sin e_3 \\ 0 \end{bmatrix} v_r + \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & -1 \end{bmatrix} u_B \tag{14}$$

식(14)를 기준궤적 주위로 선형화시키면 식(15)와 같은 선형 모델을 구할 수 있다.

$$\dot{e} = \begin{bmatrix} 0 & \omega_r & 0 \\ -\omega_r & 0 & v_r \\ 0 & 0 & 0 \end{bmatrix} e + \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & -1 \end{bmatrix} u_B \tag{15}$$

이것은 $\dot{e} = A_c e + B_c u_B$인 상태 공간 형태이다. 제어성 행렬 $[B_c\ A_c B_c\ A_c^2 B_c]$은 v_r과 ω_r이 0이 아닌 상수일 때 full rank를 갖고 이 때 제어가능이다. 이 경우에 smooth 정적 피드백으로 시스템을 안정화시키는 것이 가능하다[13].

3.3. 제어기 설계

궤적추적을 위한 제어기를 설계하기 위해 식(15)를 식(16)과 같이 이산시간으로 변환해야 한다.

$$e(k+1) = Ae(k) + Bu_B(k) \tag{16}$$

여기서 $A \in R^n \times R^n$, $B \in R^n \times R^m$이 n과 m은 각각 상태변수의 개수와 입력의 개수를 나타낸다. 이산 행렬 A, B는 식(17)과 같다.

$$A = I + A_c T_s \qquad B = B_c T_s \tag{17}$$

이것은 샘플링 시간 T_S가 짧을 때 효과적이다.

수평이동제어개념은 추정된 로봇추적 오차에 기반한 이차비용 함수를 최소화하는 제어 변수의 값을 찾는 것이다. 이차비용 함수는 식(18)과 같다.

$$J(u_B, k) = \sum_{i=1}^{h} \epsilon^T(k, i) Q \epsilon(k, i) + u_B^T(k, i) R u_B(k, i) \quad (18)$$

여기서 $\epsilon(k, i) = e_r(k+i) - e(k+i|k)$이고 $e_r(k+i)$와 $e(k+i|k)$는 각각 기준로봇추적과 실제로봇추적 오차를 의미한다. 그리고 Q와 R은 가중치 행렬을 의미하고 $Q \geq 0$, $R \geq 0$, $Q \in R^n \times R^n$, $R \in R^n \times R^m$을 만족한다.

이동하는 시간 체계에서 순간적인 시간 h에서의 모델의 출력추정은 식(19)와 같다[13].

$$\begin{aligned} e(k+h|k) = & \prod_{j=0}^{h-1} A(k+j|k) e(k) + \sum_{i=1}^{h} (\prod_{j=1}^{h-1} A(k+j|k) \quad (19) \\ & \times B(k+i-1|k) u_B(k+i-1)) \\ & + B(k+h-1|k) u_B(k+h-1) \end{aligned}$$

로봇추적 추정오차 벡터는 식(20)과 같이 정의된다.

$$E^*(k) = [e(k+1|k)^T \ e(k+2|k)^T \ \cdots \ e(k+h|k)^T]^T \quad (20)$$

여기서 $E^* \in R^{n \cdot h}$이다. 그리고 제어벡터는 식(21)과 같이 정의된다.

$$U_B(k) = [u_B(k)^T \ u_B(k+1)^T \ \cdots \ u_B(k+h-1)^T]^T \quad (21)$$

로봇추적 추정오차벡터는 식(22)와 같이 표현할 수 있다.

$$E^*(k) = F(k) e(k) + G(k) U_B(k) \quad (22)$$

여기서 $F(k)$와 $G(k)$는 식(23)과 같다.

$$F(k) = [A(k|k) \ A(k+1|k) A(k|k) \ \cdots \ \Lambda(k,0)] \quad (23)$$

$$G(k) = \begin{bmatrix} B(k|k) & 0 & \cdots & 0 \\ A(k+1|k)B(k|k) & B(k+1|k) & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \Lambda(h-1,1)B(k|k) & \Lambda(h-1,2)B(k+1|k) & \cdots & B(k+h-1|k) \end{bmatrix}$$

단, $F(k) \in R^{n \cdot h} \times R^n$, $G(k) \in R^{n \cdot h} \times R^{m \cdot h}$이고 $\Lambda(h-1,j)$는 식(24)와 같이 정의된다.

$$\Lambda(h-1,i) = \prod_{j=i}^{h-1} A(k+j|k) \quad (24)$$

이 제어법칙의 목적은 미래의 기준궤적에 실제로봇이 가능한 가깝게 주행하도록 하는 것이다. 이것은 미래의 기준궤적을 알아야한다는 것을 의미한다. 이제 상태공간에서 기준 오차 추적궤도를 식(25)와 같이 정의한다.

$$e_r(k+i) = A_r^i e(k) \quad (25)$$

기준모델 행렬(reference model matrix) A_r에 따라 미래의 제어오차가 감소해야 하는 것을 의미한다. 여기서 $i = 1, \cdots, h$이다.

로봇의 기준 추적오차 벡터는 식(26)과 같다.

$$E_r^*(k) = [e_r(k+1|k)^T \ e_r(k+2|k)^T \ \cdots \ e_r(k+h|k)^T]^T \quad (26)$$

여기서 $E_r^* \in R^{n \cdot h}$이다. 그러면 식(27)과 같은 식을 얻을 수 있다.

$$E_r^*(k) = F_r e(k) \quad (27)$$

이 때 $F_r = [A_r \ A_r^2 \ \cdots \ A_r^h]^T$ 이고 $F_r \in R^{n \cdot h} \times R^n$이다.

모델예측제어의 개념은 어떠한 추정된 구간에서 기준로봇 궤적오차와 추정된 로봇궤적과의 차이를 최소화하는 것이다. 따라서 비용함수는 식(28)과 같다.

$$J(U_B) = (E_r^* - E^*)^T \overline{Q} (E_r^* - E^*) + U_B^T \overline{R} U_B \quad (28)$$

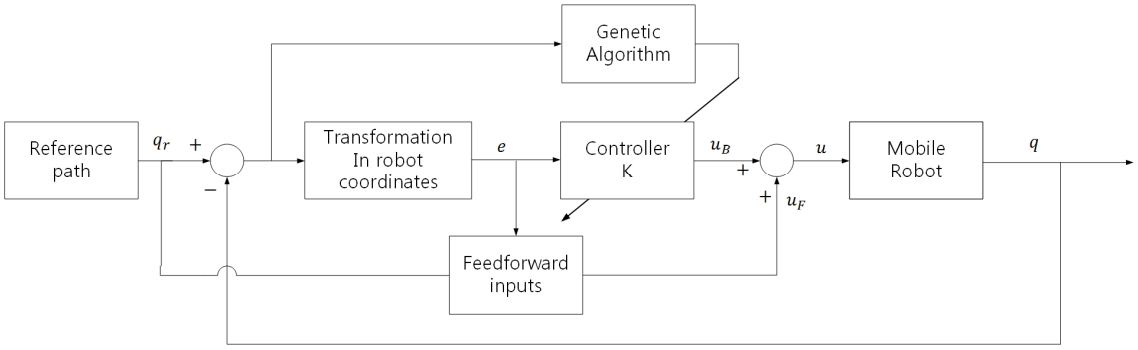


Fig. 3 Block diagram of the system

제어법칙은 비용함수를 식(29)와 같이 최소화함으로써 구할 수 있다.

$$\frac{\partial J}{\partial U_B} = -2\overline{Q}G^T + 2G^T\overline{Q}E^* + 2\overline{R}U_B = 0 \qquad (29)$$

$U_B(k)$에 대해 정리하면 식(30)과 같이 제어벡터를 구할 수 있다.

$$U_B(k) = (G^T\overline{Q}G + \overline{R})^{-1}G^T\overline{Q}(F_r - F)e(k) \qquad (30)$$

여기서

$$\overline{Q} = \begin{bmatrix} Q & 0 & \cdots & 0 \\ 0 & Q & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q \end{bmatrix} \quad \overline{Q} = \begin{bmatrix} R & 0 & \cdots & 0 \\ 0 & R & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R \end{bmatrix}$$

이고 $\overline{Q} \in R^{n \cdot h} \times R^{n \cdot h}$, $\overline{R} \in R^{m \cdot h} \times R^{m \cdot h}$을 의미한다.

3.4. 유전 알고리즘

예측제어기에서 제어성능과 관계된 제어 파라미터들이 있는데 임의로 지정되어 최적화되지 못하였다. 본 연구에서 이동로봇 예측제어기의 성능 개선을 위해 관련된 제어 파라미터들을 유전알고리즘으로 최적화하였다.

유전알고리즘은 생물의 진화를 컴퓨터상에 실현하려는 알고리즘으로 1960년대에 John Holland에 의해 발표되었다[14]. 유전 알고리즘은 초기 개체군으로 복제, 교배, 돌연변이 과정을 거친다.

유전 알고리즘은 자연계의 생물 유전학에 기반을 둔 병렬탐색 알고리즘이다. 모든 생물은 여러 가지 환경 속에 적응함으로 살아남는다는 다윈의 적자생존의 원리를 사용한다. 자연에 있는 모든 생물들은 유전자인 염색체를 통하여 유전과 생식 등 진화과정을 반복하고 발전해 간다. 유전 알고리즘에서는 컴퓨터상에서 생물학적인 염색체와 같은 문자열을 가지고 유전 연산자인 복제, 교배, 돌연변이를 거쳐서 다음 세대의 자손을 인공적으로 만들어 낸다.

유전 알고리즘은 탐색 및 최적화 기법으로 사용된다. 최적화 하고자 하는 값에 대한 모든 해들을 정해진 자료구조로 표현한 후 이를 유전 연산자를 통해 점진적으로 변형시켜 최적의 해를 구한다. 즉 $y = f(x)$를 최적화시키는 해 x의 값을 찾는 모의 진화 알고리즘이다.

유전 알고리즘은 파라미터 하나가 아닌 파라미터 집합을 코딩하고 탐색 공간 내에서 하나의 해가 아니라 해 집단을 사용한다. 또한 결정론적인 규칙이 아닌 확률적인 규칙을 사용하기 때문에 최적화 문제에서 믿을 만한 최적의 전역 해를 발견 할 가능성이 높다. 그리고 연속성, 미분가능성 등과 같은 최적화 함수의 정보를 필요로 하지 않고 오직 적합함수 값만을 이용한다.

기준모델행렬 A_r의 성분들을 유전 알고리즘의 염색체 즉 탐색변수로 설정하고 그 값에 따라 기준궤적과 로봇 위치의 차이인 오차를 최소화하는 값을 구한다. 이 때 적합도 함수 f는 식(31)과 같이 정의한다.

$$f = \sum[C_x(x_r - x)^2 + C_y(y_r - y)^2 + C_\theta(\theta_r - \theta)^2] \qquad (31)$$

여기서 C_x, C_y, C_θ는 가중치를 나타낸다. 초기의 로봇 위치가 기준궤적에 빠르게 수렴하도록 C_x의 값을 높게 설정한다. 이동로봇은 y축으로의 움직임이 힘들지만 θ에 대한 움직임은 쉬운 편이기 때문에 C_y값을 높게 설정하는 반면 C_θ의 값은 작게 설정한다.

제어시스템의 전체적인 블록선도는 그림 3과 같다.

Ⅳ. 모의실험

본 논문의 모의실험은 가상의 기준 로봇과 실제 구동로봇으로 구성하였으며 모의실험에 사용한 기구적 변수는 $m=0.6kg$, $R=3.75cm$, $r=2.5cm$ 이다. 실제 구동 로봇의 초기 위치는 $[x\ y\ \theta]=[1.1\ \ 0.7\ \ \pi/4]$이고, 가상의 기준 로봇은 식(32)와 같은 궤적을 따라 움직인다.

$$\begin{aligned} x_r &= 1.1+0.7\sin\left(\frac{2\pi t}{30}\right) \\ y_r &= 0.9+0.7\sin\left(\frac{4\pi t}{30}\right) \end{aligned} \tag{32}$$

여기서 시간 t는 0에서 30초이다.

예측제어의 제어 값은 A_r, Q, R 세 개의 파라미터에 의해 값이 변한다. A_r은 기준모델 행렬(reference model matrix)이고 Q와 R은 가중치 행렬이다. 각 행렬들은 대각행렬이다.

파라미터 값을 구하기 위해 최적화 알고리즘인 유전알고리즘을 이용하였다. 식(31)의 적합도 함수를 이용하여 미래의 제어오차의 영향을 주는 기준모델 행렬 A_r의 값을 먼저 구해보았다. 이 때 각 오차의 가중치의 값 $[C_x\ C_y\ C_\theta]$은 $[40\ 40\ 20]$, $[50\ 50\ 1]$, $[60\ 30\ 10]$ 등 여러 가지 가중치를 정해서 유전 알고리즘을 돌려보았다. 이 때 유전 알고리즘의 학습조건은 세대수 200, 개체 수 200, 교차율 0.8, 돌연변이율 0.2로 지정하였다. 또 A_r은 식(25)와 같이 오차의 값을 줄여야 하므로 값을 0과 1사이로 설정하였다.

그림 4는 유전알고리즘으로 구한 각 가중치들의 대한 최적 값을 그래프로 나타내었다. x의 가중치가 클수록 로봇의 초기응답이 빠름을 알 수 있다. 모의실험에서 사용한 가중치는 $[C_x\ C_y\ C_\theta]=[60\ 30\ 10]$이다.

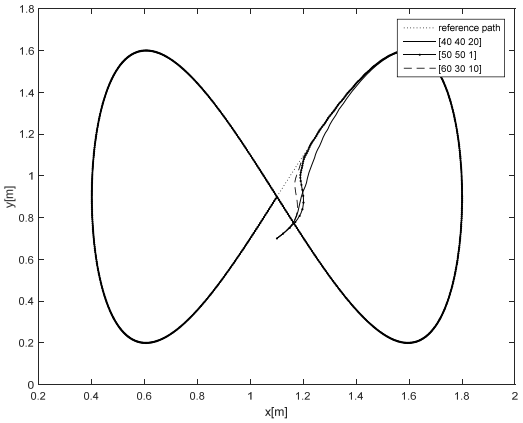
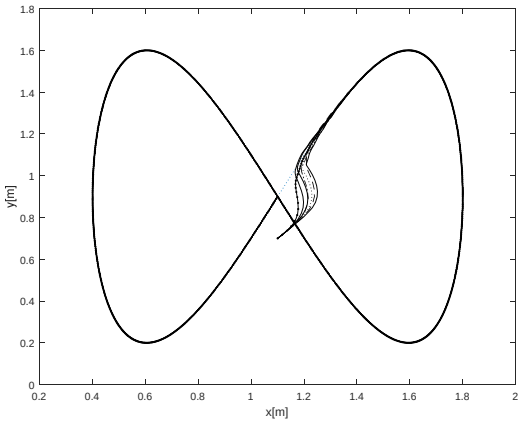
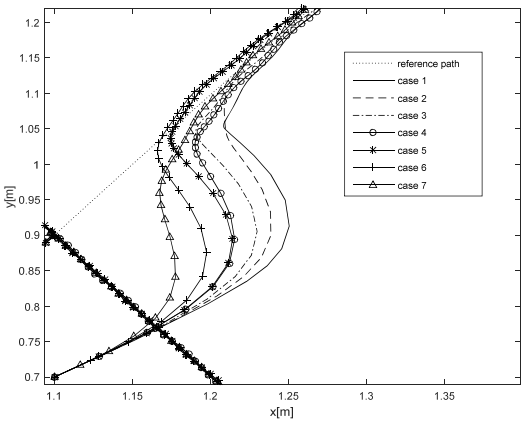


Fig. 4 Trajectory tracking with predictive controller for different weighting value

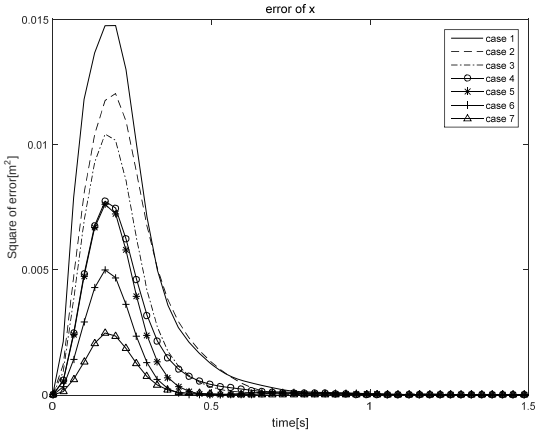


(a)

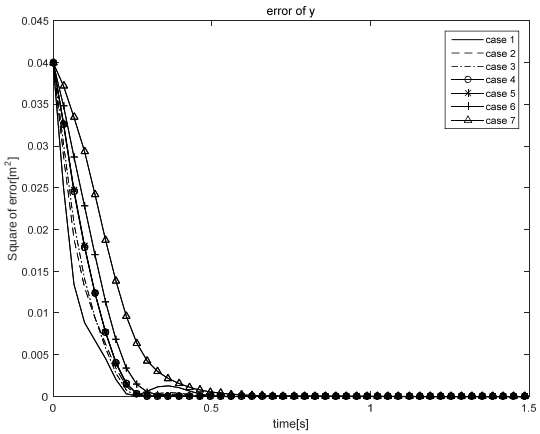


(b)

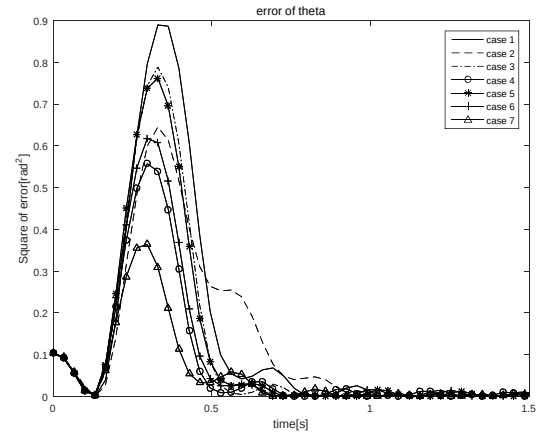
Fig. 5 (a) Trajectory tracking with predictive controller for different A_r (b) Detailed view of the initial time response



(a)



(b)



(c)

Fig. 6 (a) Errors of x-position for different A_r (b) Errors of y-position for different A_r (c) Errors of θ-angle for different A_r

그림 5 (a)는 A_r의 값이 유전알고리즘을 통해 최적화 되어가는 과정을 보여준다. 그림 5 (b)는 각 값들에 대한 초기응답을 보여준다. Case 1에서 case 7로 값이 변할수록 초기응답의 수렴하는 시기가 더 빠르다는 것을 알 수 있다.

그림 6은 그림 5의 A_r 값에 따른 기준궤적 $[x_r \ y_r \ \theta_r]$과 로봇의 위치 $[x \ y \ \theta]$의 오차의 제곱을 나타내었다. A_r 값이 case 1에서 case 7으로 바뀔 때에 y의 오차는 약간 증가하지만 x와 θ의 오차가 많이 감소함을 알 수 있다. 제어기의 성능은 로봇의 Sum of Squared Error (SSE)를 통해 확인하였다. 각 성분의 SSE 는 식(33)과 같다.

$$SSE_x = \sum (x_r - x)^2 \qquad (33)$$
$$SSE_y = \sum (y_r - y)^2$$
$$SSE_\theta = \sum (\theta_r - \theta)^2$$

Table. 1 Sum of squared error according to A_r value

Case	A_{r11}	A_{r22}	A_{r33}	SSE x	SSE y	SSE θ
1	0.2691	0.2240	0.1929	0.1165	0.1066	10.0538
2	0.4904	0.6420	0.7592	0.0955	0.1224	6.0763
3	0.5341	0.3454	0.1728	0.0692	0.1260	8.8142
4	0.6409	0.5402	0.1454	0.0527	0.1416	7.1858
5	0.6443	0.2414	0.5888	0.0450	0.1418	7.8699
6	0.7367	0.4494	0.6781	0.0283	0.1674	4.4396
7	0.8640	0.6828	0.7068	0.0139	0.2275	2.6982

표 1은 그림 5에 나와 있는 그래프들의 SSE 값을 나타낸 것이다. A_r의 각 성분들은 미래 제어 오차를 감소시키는 역할로 0에서 1사이에 값으로 나타낸다. 표1에서 보면 x의 오차는 0보다 1에 가까울수록 줄어드는 경향이 있지만 다른 성분들은 뚜렷한 경향이 나타나지 않음을 볼 수 있다. 3개의 성분이 독립적으로 영향을 미치는 것이 아님을 확인 할 수 있다. 그림 6에서 확인한 바와 마찬가지로 Case 1에서 case 7으로 바뀔수록 SSE_y는 약간 증가하지만 SSE_x와 SSE_θ는 많이 감소함을 수치적으로 확인 할 수 있다.

다음으로 가중치 행렬인 Q, R의 값을 유전알고리즘을 통해 최적제어 값을 구하였다. 유전 알고리즘에 의해 확인한 결과 Q와 R에 대해서 성능의 변화가 미미했

으며 아래와 같이 값을 지정했다.

$$Q = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 40 & 0 \\ 0 & 0 & 0.1 \end{bmatrix} \quad R = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix}$$

V. 결 론

본 연구에서는 이동 로봇의 예측제어기에 관한 연구로서 이동로봇 예측제어기 성능 개선을 위해 관련 제어 파라미터들을 최적화시키는 것에 목적이 있다. 이 때 제어 파라미터들의 값을 시행착오적으로 결정하는데 많은 어려움이 있다. 따라서 제어이득 값을 구할 때 필요로 하는 제어 파라미터들의 값을 최적화시키기 위해 전역 탐색 알고리즘으로 널리 사용되고 있는 유전알고리즘을 이용하였다.

모의실험을 통하여 예측 제어기의 성능을 결정하는 파라미터들 중 미래의 제어오차를 결정하는 기준궤적 행렬 A_r을 유전알고리즘을 이용하여 최적화시켰을 때 제어기의 성능이 좋아짐을 알 수 있었다. 그러나 가중치 행렬 Q와 R의 경우에는 유전 알고리즘으로 확인한 결과 제어기의 성능에 큰 변화가 없음을 확인 할 수 있었다.

ACKNOWLEDGMENTS

This work was supported by BK21PLUS, Creative Human Resource Development Program for IT Convergence.

REFERENCES

[1] F. Fahimi, *Autonomous Robots: Modeling, Path Planning, and Control*, New York, NY: Springer, 2010.

[2] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, *2nd ED,* Cambridge, MA: The MIT Press, 2011.

[3] T. Fukao, H. Nakagawa, and N. Adachi,"Adaptive tracking control of a nonholonomic mobile robot," *IEEE Transactions on Robotics and Automation,* vol. 16, no.5, pp. 609-615, Oct. 2000.

[4] A. K. Das et. al., "A vision-based formation control framework," *IEEE Transactions on Robotics and Automaton,* vol. 18, no. 5, pp. 813-825, Oct. 2002.

[5] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics," *Proceedings IEEE Conference on Decision and Control,* pp. 3805-3810, 1995.

[6] E. R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot using neural networks," *IEEE Transactions on Neural Networks*, vol. 9, no. 4, pp. 389-600, July 1998.

[7] Y. Wang, S. Wang, R. Tan, and Y. Jiang, "Motion control of a wheeled mobile robot using digital acceleration control method," *International Journal of Innovative Computing, Information and Control,* vol. 9, no. 1, pp. 387-396, Jan 2013.

[8] J. M. Maciejowski, *Predictive Control with Constraints*, Pearson Education Ltd., 2002.

[9] D. Gu and H. Hu, "Receding horizon tracking control of wheeled mobile robots," *IEEE Transactions on Control System Technology,* vol. 14, no. 4, pp. 743-749, July 2006.

[10] G. Klancar and I. skrjanc, "Tracking-error model-based predictive control for mobile robos in real time," *Robotics and Autonomous Systems,* vol. 55, no. 6, pp. 460-469, June 2007.

[11] Y. Gao, C. G. Lee, and K. T. Chong, "Receding horizon tracking control for wheeled mobile robots with time-delay," *Journal of Mechanical Science and Technology*, vol. 22, no. 12, pp. 2403-2416, Dec. 2008.

[12] T. P. Nascimentoa, A. P. Moreirab, and A. S. Conceicao, "Multi-robot nonlinear model formation control: Moving target and target abscence," *Robotics and Autonomous Systems,* vol. 61, no. 12, pp. 1502-1515, Dec. 2013.

[13] G. Klančar, I. Škrjanc, "Tracking-error model-based predictive control for mobile robots in real time," *Robotics and Autonomous Systems*, vol. 55, no. 6, pp.460-469, June 2007.

[14] C. T. Lin and C. S. G. Lee, *Neural Fuzzy Systems,* Upper Saddle River, NJ: Prentice Hall, 1996.

손현식(Hyun-Sik Son)

2015년 2월 부산대학교 전기전자공학과 공학사
2015년 3월 ~ 현재 부산대학교 대학원 전기전자컴퓨터공학과 석사과정
※관심분야 : 이동로봇, 지능제어, 로봇제어, 유전알고리즘 등

박진현(Jin-Hyun Park)

1992년 2월 부산대학교 전기공학과 공학사
1994년 2월 부산대학교 대학원 전기공학과 공학석사
1997년 2월 부산대학교 대학원 전기공학과 공학박사
1997년 3월 ~ 1999년 2월 부산정보대학 전기전자계열 전임강사
1996년 7월 ~ 1997년 8월 Dept. of System Engineering, Arkansas Univ. at Little Rock, Visiting Scholar
1999년 3월 ~ 현재 경남과학기술대학교 메카트로닉스공학과 교수
※관심분야 : 지능제어, 로보틱스, 제어계측 등

최영규(Young-Kiu Choi)

1980년 2월 서울대학교 전기공학과 공학사
1982년 2월 KAIST 전기 및 전자공학과 공학석사
1987년 2월 KAIST 전기 및 전자공학과 공학박사
1986년 ~ 현재 부산대학교 전기공학과 교수
※관심분야 : 로봇제어, 지능제어, 신경회로망 등