

동적마찰 섭동을 갖는 자율이동 로봇 시스템의 강인적응제어 및 안정성 해석

Robust Adaptive Control of Autonomous Robot Systems with Dynamic Friction Perturbation and Its Stability Analysis

조 현 철, 이 권 순*

(Hyun Cheol Cho and Kwon Soon Lee)

Abstract : This paper presents a robust adaptive control method using model reference control strategy against autonomous robot systems with random friction nature. We approximate a nonlinear robot system model by means of a feedback linearization approach to derive nominal control law. We construct a Least Square (LS) based observer to estimate friction dynamics online and then represent a perturbed system model with respect to approximation error between an actual friction and its estimation. Model reference based control design is achieved to implement an auxiliary control in order for reducing control error in practice due to system perturbation. Additionally, we conduct theoretical study to demonstrate stability of the perturbed system model through Lyapunov theory. Numerical simulation is carried out for evaluating the proposed control methodology and demonstrating its superiority by comparing it to a traditional nominal control method.

Keywords : autonomous robot, perturbation, random friction, robust adaptive control, stability analysis

I. 서론

자율 이동로봇 시스템은 자동화 산업의 발달과 함께 눈부시게 성장하고 있는 산업기술로서 최근 학계에서도 많은 관심을 받고 있는 공학의 한 분야이다. 특히 대규모 공장이나 복잡한 구조를 지닌 산업현장에서 자율 이동로봇은 과거에는 처리하기 어려운 작업공정에서 매우 효율적으로 사용되고 있다. 이러한 로봇 시스템은 원하는 생산 공정의 작업위치나 경로를 정확히 추적하는 정밀 제어능력을 지니고 있어야 한다. 이것은 전체 작업공정의 성능에도 직접적인 관계가 있으며 나아가 공장의 생산성 및 시스템의 효율성 면에도 깊은 연관이 있다. 최근 이러한 로봇 시스템의 정밀 제어에 많은 연구가 발표되고 있으며 우수한 적용사례가 소개되고 있다.

Harter는 진화적 신경회로망 제어를 개발하여 자율이동 로봇시스템의 경로계획 및 궤적제어에 적용하였으며[1], Ohnishi와 Asakura는 퍼지이론을 적용하여 거미로봇의 효율적인 주행속도를 위한 제어시스템을 구성한 지능형 로봇제어방식을 제안하였다[2]. Parnichkun 등은 슬라이딩 모드 제어기법을 이용하여 6자유도를 갖는 수중 탐사 로봇의 최적 제어방식을 제안하여 플랜트의 비선형성과 파라미터의 불확실성에 대한 제어성능을 개선시켰으며[3], 한편 기존의 vision 시스템 기법을 로봇의 경로제어에 적용한 제어응용 사례가 발표되었다[4]. 또한 nonholonomic 구동장치를 갖는

이동로봇의 자율제어시스템 또한 발표가 되었으며[5], 최근에는 기구학적으로 보다 복잡하고 불안정한 구조를 갖는 외바퀴 로봇시스템의 적응제어기법을 발표하였다[6].

이제까지 자율이동 로봇의 제어시스템은 로봇의 복잡한 수학적 모델을 선형화를 한 후 선형시스템 이론을 적용하여 제어를 설계하는 과정을 거치고 있다. 이러한 설계기법은 제어기 설계가 비교적 간단하고 실시간 시스템 구현이 용이하다는 특징이 있다. 하지만, 선형화로 인해 발생하는 모델링 오차나 시스템 섭동으로 인한 불확실성을 고려하지 않으므로 실시간 시스템 구현에서 예상치 못한 제어 오차가 발생할 수 있으며 결국 바람직하지 못한 제어성능을 초래할 수 있다.

특히 시스템 섭동은 오프라인 설계과정에서 예측하기 어려운 동특성이며 일반적으로 이러한 특성을 배제한 제어시스템 설계과정이 이루어지고 있다. 하지만 로봇시스템의 정밀제어와 같은 고급제어시스템 구현에 있어 이러한 요소를 고려하여야 매우 효율적인 제어성능을 기대할 수 있다. 일반적으로 시스템 섭동은 모델의 근사화나 선형화로 인하여 발생하는 모델링 오차를 들 수 있으며, 마찰력이나 비선형 특성과 같은 기구학적 모델링에서 예상치 못한 시스템 거동 등도 마찬가지로 고려할 수 있다.

본 논문은 로봇시스템의 마찰력으로 인한 시스템섭동을 보상하기 위한 강인적응제어 시스템을 제안한다. 우선, 종래의 방법과 마찬가지로 로봇시스템의 비선형모델을 선형화하여 공칭제어규칙을 산출한다. 다음으로, 시스템 섭동모델에 대하여 모델기반 적응제어기법을 이용하여 보조제어 시스템을 구성하며, 이 때 시스템 섭동은 마찰력 추정값과 실제 마찰력에 대한 추정오차로 표현되어진다. 또한 시스템 섭동모델에 대하여 Lyapunov 이론을 적용하여 안정성 해석

* 책임저자(Corresponding Author)

논문접수 : 2008. 9. 6. 채택확정 : 2008. 12. 8.

조현철 : 동아대학교 전기공학과(hyunccho@gmail.com)

이권순 : 동아대학교 전기공학과(kslee@dau.ac.kr)

※ 본 논문은 2008학년도 동아대학교 학술연구비에 의하여 연구되었음.

을 실시하며 안정하기 위한 시스템의 필요충분조건을 제시한다. 제안한 제어기법의 타당성을 검증하기 위하여 컴퓨터 시뮬레이션을 실시하였으며 성능의 우수성을 입증하기 위하여 기존의 공칭제어기법과 비교 및 검토하였다.

본 논문의 구성은 다음과 같다. II 장에서는 본 논문에서 다루는 자율이동 로봇시스템의 모델에 대하여 서술하며, III 장에서 비선형 시스템모델에 대한 선형화를 실시하며, IV 장에서 LS 기반 마찰력 추정알고리즘을 제안한다. V 장에서 시스템 섭동모델을 서술하며, VI 장에서 모델기반 적응 제어규칙을 산출하며, VII 장에서 시스템 안정성 해석을 실시하며, VIII 장에서 컴퓨터 시뮬레이션에 대한 결과 및 검토를, 마지막 IX 장에서 결론 및 향후 연구계획을 서술한다.

II. 자율이동 로봇시스템 모델

본 논문에서 다루는 자율이동 로봇은 그림 1과 같은 4륜 구동방식의 시스템으로서, 로봇의 액츄에이터로서 DC 전동기가 바퀴에 부착이 되어있으며 내부에 임베디드 마이크로 컨트롤러가 장착이 되어있으며 시스템 프로그램에 의하여 자율적으로 구동되어진다. 그림 2는 이 시스템의 기구학적 모델과 이동방향에 대한 좌표를 보여준다. 로봇의 움직임은 2차 평면의 원점 $\{O, X, Y\}$ 을 기준으로 하여 로봇의 무게중심의 방향좌표 $\{c, x_c, y_c\}$ 로 표현되어진다. 따라서 X 축과 Y 축에 대한 로봇의 실제 이동거리는 x_c 와 y_c 그리고 원점에 대한 이동각도를 θ 를 이용하여 표현된다. 로봇시스템의 동특성은 Lagrange 방정식의 형태로 주어지며 다음과 같다[7].

$$M(q)\ddot{q} + V(q, \dot{q}) + F(\dot{q}) = B(q)\tau - A^T(q)\lambda \quad (1)$$

여기서 q 는 시스템 상태벡터, τ 는 시스템 입력벡터, 그리고 λ 는 제약입력 벡터를 각각 나타낸다. 또한, 대칭 및 positive definite 구조를 갖는 $M(q)$ 는 시스템의 관성을 표현하는 행렬이며, $V(q, \dot{q})$ 는 centripetal 및 coriolis 행렬, $F(\dot{q})$ 은 표면 마찰력 특성을 내포하고 있으며, $B(q)$ 은 입력행렬, $A(q)$ 은 제약입력에 대한 행렬을 각각 나타낸다. Lagrange 모델링기법을 이용하여 로봇의 동적 방정식을 표현할 수 있으며, 식 (1)에서 로봇은 단지 수평방향의 움직임만을 고려할 경우 위치에너지는 일정하게 되며 운동방정식 K 는 다음과 같이 주어진다[7].

$$K = \sum_{i=1}^n \left\{ \frac{1}{2} m_i v_i^T + \frac{1}{2} w_i^T I_i w_i \right\} = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (2)$$

따라서 식 (2)의 운동방정식에 대하여 식 (1)의 각각의 행렬과 벡터변수는 다음과 같이 정의되어진다.

$$M(q) = \begin{bmatrix} m & 0 & m d \sin \theta \\ 0 & m & -m d \cos \theta \\ m d \sin \theta & -m d \cos \theta & I \end{bmatrix},$$

$$V_m(q, \dot{q}) = \begin{bmatrix} 0 & 0 & m d \dot{\theta} \cos \theta \\ 0 & 0 & m d \dot{\theta} \sin \theta \\ 0 & 0 & 0 \end{bmatrix}, \quad G(q) = \mathbf{0},$$

$$B(q) = \frac{1}{r} \begin{bmatrix} \cos \theta & \cos \theta \\ \sin \theta & \sin \theta \\ R & -R \end{bmatrix}, \quad \tau = \begin{bmatrix} \tau_r \\ \tau_l \end{bmatrix}, \quad A^T(q) = \begin{bmatrix} -\sin \theta \\ \cos \theta \\ -d \end{bmatrix},$$

$$\lambda = -m(\dot{x}_c \cos \theta + \dot{y}_c \sin \theta) \dot{\theta}$$

여기서 m 은 로봇의 질량을 나타내며, 위의 식으로부터 식 (1)의 시스템 모델은 매우 복잡한 비선형 관계로 이루어짐을 알 수 있다.

III. 비선형 로봇 시스템의 피드백 선형화

식 (1)의 비선형 시스템 모델은 Lagrange 형태의 방정식이므로 피드백 선형화 기법을 적용하여 제어를 구현할 수 있다. 식 (1)에서 외란을 고려하지 않은 상태에서 즉 $\tau_d = 0$ 을 두고, 제어입력항인 $B\tau$ 를 다음과 같이 정의한다.

$$B\tau = Mu + V + F + A^T \lambda \quad (3)$$

또는 식 (3)을 τ 에 대하여 정리하면

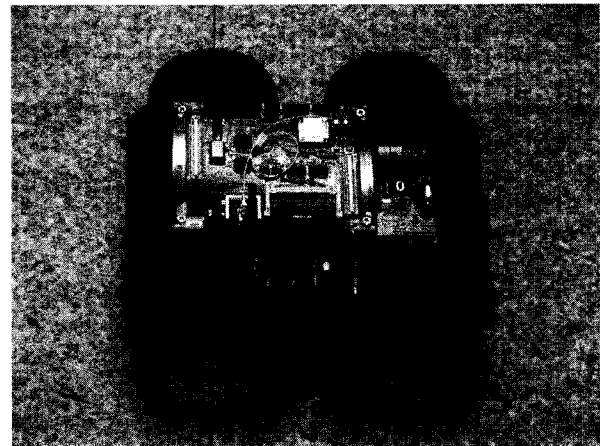


그림 1. 자율이동로봇시스템의형. Fig. 1. Appearance of an autonomous robot system.

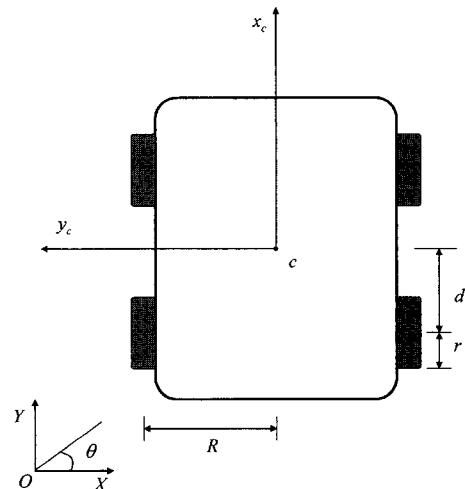


그림 2. 자율이동 로봇의 기구학적 모델. Fig. 2. A geometric model of an autonomous robot.

$$\tau = B^+(Mu + V + F + A^T\lambda) \quad (4)$$

가 되며 여기서 B^+ 는 행렬 B 의 pseudo 역행렬로서 $B^+ = (B^T B)^{-1} B^T$ 로 주어지며 u 는 새로운 제어입력변수를 나타낸다. 식 (3)을 식 (1)에 대입하여 전개하면 다음과 같은 선형시스템 모델로 근사화된다.

$$\ddot{q}(t) = u(t) \quad (5)$$

여기서 u 를 다음과 같은 선형제어기로 정의한다.

$$u = K_1 \dot{q} + K_2 \ddot{q} + r \quad (6)$$

여기서 $K_1, K_2 \in R^{3 \times 3}$ 은 제어파라미터 행렬을 나타낸다. 식 (6)을 식 (5)에 대입하면 시스템 모델은 최종적으로 다음과 같이 전개된다.

$$\ddot{q} - K_2 \dot{q} - K_1 q = r \quad (7)$$

식 (7)은 전형적인 2계 미분방정식으로서 기존의 잘 알려진 선형시스템 이론을 적용하여 안정하고 적절한 제어사양을 만족하기 위한 파라미터 행렬 K_1 과 K_2 를 결정할 수 있다.

IV. 슬라이딩 윈도우 기반 LS 마찰 관측기

식 (7)의 선형화 모델은 식 (3)에서 마찰력 변수 F 가 정확하다는 가정하에 산출된 것으로서 실질적으로 마찰력은 이론적인 모델과 차이가 존재하므로 식 (7)의 이상적인 방정식에 대하여 모델링 오차가 발생할 수 있다. 따라서 이러한 경우 해석적 모델을 시스템 방정식에 두는 것보다는 실제 시스템 관측을 통해 마찰력을 추정하는 것이 보다 더 효과적이다[8]. 이러한 추정은 마찰력 관측기를 이용하여 실현할 수 있으며 본 절에는 LS 기법을 이용한 온라인 마찰력 관측기를 제안한다. 우선 마찰력 추정변수 \hat{F} 를 식 (4)의 제어입력규칙에 적용하여 다시 표현하면 다음과 같으며

$$\tau = B^+(Mu + V + \hat{F} + A^T\lambda) \quad (8)$$

식 (1)의 방정식에 대입한 후 변수분리법을 적용하여 나타내면 다음과 같다.

$$\begin{cases} \dot{q}_1 = q_2 \\ \dot{q}_2 = M^{-1}\{B\tau - A^T\lambda - V - \hat{F}\} \end{cases} \quad (9)$$

식 (9)의 연속시간 변수를 이산시간 k 에 대한 변수로 나타내기 위하여, 미분의 근사화 공식을 적용하여 $\dot{q} \approx \Delta q(k)/h = \{q(k) - q(k-1)\}/h$ 와 같이 정의하면 다음과 같다.

$$\begin{cases} \Delta q_1(k) = h q_2(k) \\ \Delta q_2(k) = -h M^{-1}\{B\tau - A^T\lambda - V - \hat{F}\} \end{cases} \quad (10)$$

여기서 h 는 샘플링 상수이다. 식 (10)의 마찰력 관측기를 구현하기가 용이한 다항식 모델로 정의하며 다음과 같다 [8].

$$\hat{F} = \sum_{i=1}^n \theta_i q_2^{i-1}(k) \quad (11)$$

식 (11)를 식 (10)의 두 번째 식의 마찰력 변수에 대입하면

$$\Delta q_2(k) = -h M^{-1}\left\{B\tau - A^T\lambda - V - \sum_{i=1}^n \theta_i q_2^{i-1}(k)\right\} \quad (12)$$

와 같이 전개되며, 또한 식 (10)의 첫 번째 수식으로 부터 $q_2 = \Delta q_1/h$ 이므로 식 (12)는 최종적으로 다음과 같다.

$$\Delta q_2(k) = -h M^{-1}\left\{B\tau - A^T\lambda - V - \sum_{i=1}^n \theta_i \left(\frac{\Delta q_1(k)}{h}\right)^{i-1}\right\} \quad (13)$$

한편 $\Delta q_2(k) = (\Delta q_1(k) - \Delta q_1(k-1))/h$ 의 관계를 식 (13)에 적용하면 다음과 같으며

$$\begin{aligned} \Delta q_1(k) - \Delta q_1(k-1) \\ = -h^2 M^{-1}\left\{B\tau - A^T\lambda - V - \sum_{i=1}^n \theta_i \left(\frac{\Delta q_1(k)}{h}\right)^{i-1}\right\} \end{aligned} \quad (14)$$

관측변수에 대하여 정리하면 최종적으로 다음과 같이 전개된다.

$$\begin{aligned} \frac{M[\Delta q_1(k-1) - \Delta q_1(k)]}{h^2} - B\tau + A^T\lambda + V \\ = \sum_{i=1}^n \theta_i \left(\frac{\Delta q_1(k)}{h}\right)^{i-1} \end{aligned} \quad (15)$$

LS 추정알고리즘에 의하여 식 (15)의 마찰모델의 파라미터 벡터 $\theta = [\theta_1, \theta_2, \dots, \theta_n]^T$ 에 대한 추정식은 다음과 같다.

$$\hat{\theta} = (H^T H)^{-1} H^T Z \quad (16)$$

여기서 관측값과 시스템 파라미터를 포함하는 행렬 H 와 Z 는 각각 다음과 같이 주어진다.

$$H = \begin{bmatrix} 1 & \frac{\Delta q_1(1)}{h} & \dots & \left(\frac{\Delta q_1(1)}{h}\right)^{n-1} \\ 1 & \frac{\Delta q_1(2)}{h} & \dots & \left(\frac{\Delta q_1(2)}{h}\right)^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \frac{\Delta q_1(N)}{h} & \dots & \left(\frac{\Delta q_1(N)}{h}\right)^{n-1} \end{bmatrix} \quad (17)$$

$$Z = \frac{M}{h^2} \begin{bmatrix} \Delta q_1(0) - \Delta q_1(1) \\ \Delta q_1(1) - \Delta q_1(2) \\ \vdots \\ \Delta q_1(N-1) - \Delta q_1(N) \end{bmatrix} - \begin{bmatrix} B(1)\tau(1) \\ B(2)\tau(2) \\ \vdots \\ B(N)\tau(N) \end{bmatrix} \quad (18)$$

$$+ \begin{bmatrix} A^T(1)\lambda(1) \\ A^T(2)\lambda(2) \\ \vdots \\ A^T(N)\lambda(N) \end{bmatrix} + \begin{bmatrix} V(1) \\ V(2) \\ \vdots \\ V(N) \end{bmatrix}$$

여기서 행렬 H 가 full-rank가 아닐 경우 식 (16)을 Moore-Penrose pseudoinverse[9]로 대체하여 연산할 수 있다. 이 추정알고리즘은 이산시간 k 동안의 모든 관측 데이터를 사용하는 것으로 유한개의 데이터 집합을 갖는 오프라인 연산에 적합하다. 하지만, 실시간 구현에 있어 현재시간 k

이전의 모든 데이터 집합이 항상 필요한 것은 아니며, 다시 말해, 유한개의 관측 데이터만으로도 충분히 파라미터 추정 에 적용할 수 있다. 이러한 방법은 이전의 데이터 값을 저장하는 메모리 공간을 절약하는 뿐만 아니라 연산속도도 개선시킬 수가 있다. 본 논문은 식 (16)의 추정규칙에 슬라 이딩 윈도우 $N_w > 0$ 을 적용함으로써 적절한 데이터의 양 을 결정할 수 있다. 식 (17)과 식 (18)에 윈도우 변수 N_w 를 적용하여 다시 나타내면 다음과 같다.

$$H = \begin{bmatrix} 1 & \frac{\Delta q_1(k-N_w+1)}{h} & \dots & \left(\frac{\Delta q_1(k-N_w+1)}{h}\right)^{n-1} \\ 1 & \frac{\Delta q_1(k-N_w+2)}{h} & \dots & \left(\frac{\Delta q_1(k-N_w+2)}{h}\right)^{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \frac{\Delta q_1(k)}{h} & \dots & \left(\frac{\Delta q_1(k)}{h}\right)^{n-1} \end{bmatrix} \quad (19)$$

$$Z = \frac{M}{h^2} \begin{bmatrix} \Delta q_1(k-N_w) - \Delta q_1(k-N_w+1) \\ \Delta q_1(k-N_w+1) - \Delta q_1(k-N_w+2) \\ \vdots \\ \Delta q_1(k-1) - \Delta q_1(k) \end{bmatrix} - \begin{bmatrix} B(k-N_w+1)\tau(k-N_w+1) \\ B(k-N_w+2)\tau(k-N_w+2) \\ \vdots \\ B(k)\tau(k) \end{bmatrix} + \begin{bmatrix} A^T(k-N_w+1)\lambda(k-N_w+1) \\ A^T(k-N_w+2)\lambda(k-N_w+2) \\ \vdots \\ A^T(k)\lambda(k) \end{bmatrix} + \begin{bmatrix} V(k-N_w+1) \\ V(k-N_w+2) \\ \vdots \\ V(k) \end{bmatrix} \quad (20)$$

V. 시스템 섭동 모델

앞서 언급한 바와 같이 마찰력 추정알고리즘은 실시간 시스템 구현에서 추정오차를 피할 수 없으며 정밀한 제어 성능을 위하여 이러한 오차를 고려한 제어시스템의 설계가 이루어져야 한다. 우선, 미지의 실제 마찰력과 추정값에 대한 오차를 다음과 같이 정의한다.

$$\Delta F = F - \hat{F} \quad (21)$$

식 (21)의 관계식을 식 (4)에 적용하면 다음과 같으며

$$\tau = B^+(Mu + V + (F - \Delta F) + A^T\lambda) \quad (22)$$

따라서 식 (5)에 대한 시스템 섭동모델은 다음과 같이 표현 된다.

$$\ddot{q}(t) + \Delta F = \tilde{u}(t) \quad (23)$$

여기서 제어변수 $\tilde{u}(t)$ 를 식 (6)의 공칭 제어규칙과 보조제

어변수 $\Delta u(t)$ 의 선형조합으로 다시 구성하면 다음과 같다.

$$\begin{aligned} \tilde{u}(t) &= u(t) + \Delta u(t) \\ &= K_1q + K_2\dot{q} + r + \Delta u(t) \end{aligned} \quad (24)$$

식 (24)의 두 번째 항을 식 (23)에 대입하여 전개하면 다음과 같은 섭동모델로 표현된다.

$$\ddot{q} - K_2\dot{q} - K_1q + \Delta F = r + \Delta u(t) \quad (25)$$

또한 상태공간 방정식으로 표현하기 위하여 $x = [q, \dot{q}]^T$ 로 정의하여 변수분리법을 적용하여 식 (25)를 다음과 같이 표현할 수 있다.

$$\dot{x} = f(x) = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} q_2 \\ K_2q_2 + K_1q_1 - \Delta F + r + \Delta u \end{bmatrix} \quad (26)$$

VI. 선형모델기반 적응제어기 설계

식 (26)의 보조 제어입력 Δu 는 마찰력 추정오차로 인해 발생하는 시스템 섭동을 보상하기 위한 제어입력으로, 직관 적으로 $\Delta u = \Delta F$ 이면 식 (7)과 같은 이상적인 모델로 되어 제어기 구성이 쉽지만 일반적으로 ΔF 는 미지의 변수이므로 Δu 를 직접적으로 정의하기가 어렵다. 따라서 Δu 는 ΔF 에 대한 함수로 정의되는 적응제어전략을 통하여 구성 되어야 한다. 본 논문은 모델기반 적응제어기법을 이용하여 보조제어시스템 Δu 를 정의한다. 즉, 식 (26)의 시스템 동 특성이 다음의 기준모델에 추종되기 위한 제어규칙을 산출 하는 것이다.

$$\dot{\zeta}(t) = \Phi\zeta(t) + \Gamma r(t) \quad (27)$$

여기서 기준 상태변수 ζ , 기준입력 r , 그리고 기준 상태행 렬 Φ 및 입력행렬 Γ 의 크기는 식 (26)의 실제 상태방정식 과 동일한 크기로 정의되어진다. 특히 Φ 는 안정한 행렬로 설정되어야 하며, 즉 Φ 의 모든 고유치의 실수부분이 음 수가 되어야 한다. 식 (26)의 상태가 식 (27)의 상태로 추종 되는 것은 다음의 오차가 0으로 되는 것을 의미한다.

$$e(t) = \zeta(t) - x(t) \quad (28)$$

따라서 기준모델 적응제어규칙은 식 (28)의 오차방정식을 줄이기 위한 제어규칙을 산출하는 것이다. 본 논문은 Lyapunov 안정성 이론을 적용하여 이러한 제어규칙을 산출 한다. 우선 식 (28)의 오차를 이용하여 Lyapunov 함수를 다 음과 같이 정의한다.

$$V(e) = e^T(t)Pe(t) \quad (29)$$

여기서 V 가 항상 양의 값을 갖도록 정방행렬 P 를 positive definite으로 설정한다. 식 (29)의 미분식은 다음과 같이 전 개되며

$$\dot{V}(e) = \dot{e}^T(t)Pe(t) + e^T(t)\dot{P}e(t) \quad (30)$$

여기서 식 (26)과 식 (27)을 바탕으로 오차의 미분식은 다

음과 같이 전개된다.

$$\begin{aligned}\dot{e}(t) &= \dot{\zeta}(t) - \dot{x}(t) \\ &= \Phi\zeta(t) + \Gamma r - f(x) \\ &= \Phi\zeta(t) - \Phi x(t) + \Phi x(t) - f(x) + \Gamma r \\ &= \Phi e(t) + \Phi x(t) - f(x) + \Gamma r\end{aligned}\quad (31)$$

식 (31)의 결과식을 식 (30)에 대입하여 전개하면 정의한 Lyapunov 함수의 미분식은 최종적으로 다음과 같다.

$$\begin{aligned}\dot{V} &= [\Phi e + \Phi x - f(x) + \Gamma r]^T P e \\ &\quad + e^T P [\Phi e + \Phi x - f(x) + \Gamma r] \\ &= e^T (\Phi^T P + P \Phi) e + 2e^T P [\Phi x - f(x) + \Gamma r]\end{aligned}\quad (32)$$

Lyapunov 안정성 이론에 의하면 식 (28)의 오차가 수렴하기 위해서는 식 (32)의 미분값이 음수가 되어야 하며 다음과 같은 조건이 주어진다.

$$e^T (\Phi^T P + P \Phi) e + 2e^T P [\Phi x - f(x) + \Gamma r] < 0 \quad (33)$$

또한, 식 (33)의 부등식을 만족하기 위하여 다음의 두 필요충분조건을 만족하여야 한다.

$$e^T (\Phi^T P + P \Phi) e = -Q \quad (34a)$$

$$e^T P [\Phi x - f(x) + \Gamma r] < 0 \quad (34b)$$

우선, 식 (34a)에서 두 행렬 Q 와 P 가 정방행렬로서 positive definite로서 정의되어질 경우 상태행렬 Φ 을 안정한 행렬로 결정함으로써 등식은 성립되어진다. 다시 말해, 이러한 조건은 식 (27)의 기준모델에서 안정한 상태행렬 Φ 를 결정함으로써 식 (34a)의 조건은 간단히 만족되어진다. 다음으로, 식 (34b)에서 함수 $f(x)$ 는 보조제어변수 Δu 를 포함하고 있으며, 식 (34b)의 조건을 항상 만족할 수 있는 Δu 의 제어규칙을 설정하여야 한다. 우선, Lyapunov 행렬 P 를 가장 간단한 positive definite 형태인 단위행렬로 정의하면, 즉 $P = I_6$ 이면 식 (34b)에서

$$e^T P = e^T \quad (35)$$

가 된다. 다음으로 식 (27)의 기준모델을 가제어성(controlability) 형태로 정의하면 기준 제어 및 입력 행렬은 각각 다음과 같으며

$$\Phi = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \phi_1 & \phi_2 & \phi_3 & \phi_4 & \phi_5 & \phi_6 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (36)$$

우선 식 (34b)에서 Φx 와 Γr 를 다음과 같이 정의하며

$$\alpha := \Phi x, \quad \beta := \Gamma r \quad (37)$$

따라서 식 (34b)는 다음과 같이 다시 표현된다.

$$e^T P [\alpha - f(x) + \beta] < 0 \quad (38)$$

여기서 α 와 β 는 식 (36)을 적용하여 전개하면 다음과 같다.

$$\alpha = \begin{bmatrix} x_2 & x_3 & x_4 & x_5 & x_6 & \sum_{i=1}^6 \phi_i x_i \end{bmatrix}^T, \quad \beta = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \sum_{i=1}^3 r_i \end{bmatrix}^T \quad (39)$$

식 (38)에 있는 벡터의 크기는 6행 1열이며 연산의 용이성을 위하여 e , α , 그리고 β 를 3행 1열의 두 벡터로 분할하여 다음과 같이 나타낸다.

$$e = [e_1 \ e_2]^T, \quad \alpha = [\alpha_1 \ \alpha_2]^T, \quad \beta = [\beta_1 \ \beta_2]^T \quad (40)$$

식 (40)과 식 (26)를 식 (38)에 대입하여 전개하면 다음과 같다.

$$\begin{aligned} & \left[e_1^T \ e_2^T \right] \left\{ \begin{bmatrix} \Phi_1 \\ \Phi_2 \end{bmatrix} - \begin{bmatrix} K_2 q_2 + K_1 q_1 - \Delta F + r + \Delta u \\ \end{bmatrix} + \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \end{bmatrix} \right\} \\ & = e_1^T (\Phi_1 - q_2 + \Gamma_1) \\ & \quad + e_2^T (\Phi_2 - K_2 q_2 - K_1 q_1 + \Delta F - r - \Delta u + \Gamma_2) < 0\end{aligned}\quad (41)$$

여기서 보조제어변수 Δu 를 다음과 같이 정의한 후

$$\Delta u = \Delta u_1 + \Delta u_2 \quad (42)$$

식 (41)에 대입하면 두 번째 수식은 다음과 같이 전개된다.

$$\begin{aligned} & e_1^T (\Phi_1 - q_2 + \Gamma_1) \\ & \quad + e_2^T (\Phi_2 - K_2 q_2 - K_1 q_1 + \Delta F - r - \Delta u_1 - \Delta u_2 + \Gamma_2) < 0\end{aligned}\quad (43)$$

여기서 Δu_1 을 정의하기 위하여 다음과 같이 두면

$$e_2^T \Delta u_1 = e_1^T (\Phi_1 - q_2 + \Gamma_1) + e_2^T (\Phi_2 - K_2 q_2 - K_1 q_1 - r + \Gamma_2) \quad (44)$$

가 되며 따라서 다음과 같은 제어규칙이 주어진다.

$$\Delta u_1 = (e_2^T)^+ \left\{ e_1^T (\Phi_1 - q_2 + \Gamma_1) + e_2^T (\Phi_2 - K_2 q_2 - K_1 q_1 - r + \Gamma_2) \right\} \quad (45)$$

여기서 $(e_2^T)^+$ 는 e_2^T 의 pseudo 역행렬을 나타낸다. 식 (44)를 식 (43)에 대입하면 다음과 같이 줄어든다.

$$e_2^T (-\Delta u_2 + \Delta F) < 0 \quad (46)$$

여기서 각각의 벡터 요소(elements)는 다음과 같으며

$$e_2 = [e_{2,1}, e_{2,2}, e_{2,3}], \quad \Delta u = [\Delta u_{2,1}, \Delta u_{2,2}, \Delta u_{2,3}] \quad (47)$$

$$\Delta F = [\Delta F_{2,1}, \Delta F_{2,2}, \Delta F_{2,3}]$$

각각의 요소를 식 (46)에 대입하여 전개하면 다음과 같다.

$$\sum_{i=1}^3 e_{2,i} (\Delta F_i - \Delta u_{2,i}) < 0 \quad (48)$$

식 (48)의 부등식이 성립되기 위한 충분조건은 다음과 같으며

$$e_{2,i} (\Delta F_i - \Delta u_{2,i}) < 0, \quad i = 1, 2, 3 \quad (49)$$

또한 식 (49)의 조건을 만족하기 위한 제어변수는 다음과 같다.

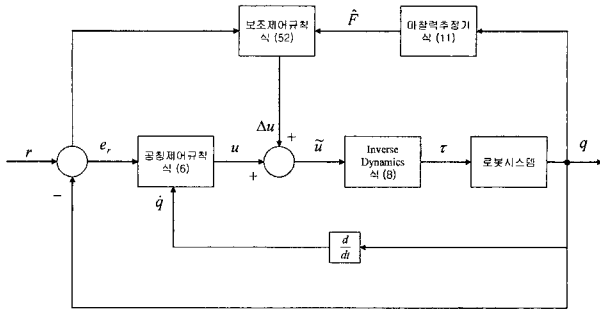


그림 3. 로봇시스템의 적응제어 블록선도.

Fig. 3. A block diagram of adaptive robot control system.

$$\Delta u_{2,i} = \text{sign}(e_{2,i}) \cdot \max|\Delta F_i|, \quad i = 1, 2, 3 \quad (50)$$

여기서 ΔF 에 대한 정보는 절대값의 최대치는 알 수 있다고 가정한다. 식 (50)을 식 (49)에 대입하면

$$e_{2,i}(\Delta F_i - \text{sign}(e_{2,i}) \cdot \max|\Delta F_i|) < 0, \quad i = 1, 2, 3 \quad (51)$$

와 같으며 식 (51)은 항상 음수를 가지므로 이 부등식은 항상 만족되어짐을 알 수 있다. 최종적으로 식 (42)의 제어규칙은 식 (45)와 식 (50)으로부터 다음과 같이 정의되어진다.

$$\Delta u = \begin{pmatrix} e_2^T \left(\Phi_1 - q_2 + \Gamma_1 \right) \\ + e_2^T \left(\Phi_2 - K_2 q_2 - K_1 q_1 - r + \Gamma_2 \right) \end{pmatrix} + \text{sign}(e_2) \cdot \max|\Delta F| \quad (52)$$

식 (52)의 보조제어규칙은 실시간 제어오차와 마찰력 추정 오차의 함수로 구성되어지며 이러한 오차를 감소시키기 위한 적응형 보상기로 작동되어진다. 그림 3은 로봇시스템의 전체 제어시스템에 대한 블록선도를 보여준다.

VII. 시스템 안정성 해석

본 절에서 식 (26)의 섭동변수를 갖는 시스템에 대한 안정성 해석을 실시하며 또한 안정하기 위한 조건을 해석적 방법을 통하여 제시한다. 본 논문은 Lyapunov 섭동이론을 통해 안정성을 분석하며 우선 증명에 관련된 이론들을 서술한다.

정리 1[10]: 원점에 평형 또는 수렴특성을 갖는 비선형 자유 시스템 $\dot{x} = f(x, t)$ 을 고려해보자. 여기서 $x \in R^n$ 의 궤적(trajjectory)은 양수 r 보다 항상 작으며 해(solution)의 궤적이 다음의 부등식을 만족한다고 가정한다.

$$\|x(t)\| \leq k \|x(t_0)\| e^{-\lambda(t-t_0)}, \quad 0 \leq t_0 \leq t \quad (53)$$

여기서 $k > 0$ 이며, Lyapunov 함수 $V(x, t)$ 에 대하여 상태 x 는 다음의 관계식을 만족한다.

$$c_1 \|x\|^2 \leq V(x, t) \leq c_2 \|x\|^2 \quad (54)$$

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(x, t) \leq -c_3 \|x\|^2 \quad (55)$$

$$\left\| \frac{\partial V}{\partial x} \right\| \leq c_4 \|x\| \quad (56)$$

여기서 $c_1, c_2, c_3, c_4 > 0$ 이다. 시스템 섭동을 갖는 비선형 모델에 대한 Lyapunov 안정성은 정리 1을 이용하여 해석할 수 있다.

한편, 공칭함수 $f(x, t)$ 와 섭동함수 $g(x, t)$ 로 표현되는 시스템 모델 $\dot{x} = f(x, t) + g(x, t)$ 에 대하여 섭동함수의 궤적이 Lipschitz 조건을 만족한다면 다음과 같은 관계로 나타낼 수 있다.

$$\|g(x, t)\| \leq \gamma \|x(t)\|, \quad \gamma > 0 \quad (57)$$

섭동시스템에 대한 Lyapunov 안정성이론에 의하면 공칭함수 $f(x, t)$ 가 식 (54)-식 (56)의 관계를 모두 만족하고 섭동함수 $g(x, t)$ 가 식 (57)의 조건을 만족하면, 섭동 시스템은 원점에서 점근적 안정(asymptotically stable)하다고 알려져 있다[10].

본 논문은 이러한 안정성이론을 이용하여 식 (26)의 섭동시스템에 대한 안정성과 그 조건을 분석한다. 우선, 식 (26)의 시스템 방정식을 공칭과 섭동부분으로 나눈 자유시스템 모델로 표현하면 다음과 같다.

$$\dot{x}(t) = Ax(t) + \Delta A(t)x(t) \quad (58)$$

여기서 공칭 상태행렬 A 는 공칭제어 파라미터 K_1 과 K_2 로 구성되어지며 다음과 같다.

$$A = \begin{bmatrix} 0 & 1 \\ K_1 & K_2 \end{bmatrix} \quad (59)$$

또한 섭동 상태행렬 $\Delta A(t)$ 는 마찰력 추정오차 ΔF 에 대한 함수로 주어지며 ΔF 는 상태 $x(t)$ 에 대한 함수이므로 최종적으로 다음과 같이 표현된다.

$$\Delta A = g(\Delta F(x, t)) \quad (60)$$

앞서 설명한 바와 같이 공칭상태행렬 A 는 식 (59)의 K_1 과 K_2 에 의하여 안정하게 설정되므로 Lyapunov 안정성 이론에 의하여 다음의 Lyapunov 방정식에 대한 해가 존재한다.

$$PA + A^T P = -Q \quad (61)$$

여기서 P 와 Q 는 positive definite으로 설정된다. 또한 Lyapunov 함수 $V(x) = x^T P x$ 에 대하여 식 (54)-식 (56)의 관계식을 구하면 각각 다음과 같다.

$$\lambda_{\min}(P) \|x\|_2^2 \leq V(x) \leq \lambda_{\max}(P) \|x\|_2^2 \quad (62)$$

$$\frac{\partial V}{\partial x} Ax = -x^T Q x \leq -\lambda_{\min}(Q) \|x\|_2^2 \quad (63)$$

$$\begin{aligned} \left\| \frac{\partial V}{\partial x} \right\|_2 &= \|2x^T P\|_2 \leq 2\|P\|_2 \|x\|_2 \\ &= 2\lambda_{\max}(P) \|x\|_2 \end{aligned} \quad (64)$$

여기서 λ 는 고유치를 나타내며 λ_{\min} 와 λ_{\max} 는 최소 및 최대 고유치를 각각 나타낸다. 또한 식 (62)-식 (64)로부터 미분식은 다음과 같이 전개된다.

$$\dot{V}(x) = -\lambda_{\min}(Q) \|x\|_2^2 + 2\lambda_{\max}(P) \gamma \|x\|_2^2 \quad (65)$$

이 관계로부터 $\dot{v}(x) < 0$ 이기 위한 필요충분조건은 다음과 같이 주어진다.

$$\gamma < \frac{\lambda_{\min}(Q)}{2\lambda_{\max}(P)} \quad (66)$$

간단하게, Q 를 단위행렬로 정의하면, 식 (66)은 다음과 같다.

$$\gamma < \frac{1}{2\lambda_{\max}(P)} \quad (67)$$

한편, 식 (57)의 관계식으로부터 식 (58)의 섭동항에 대한 안정조건은 다음과 같다.

$$\|\Delta A(t)x(t)\| \leq \gamma \|x(t)\| \quad (68)$$

여기서 행렬 norm의 특성에 의하여 다음과 같이 전개되며

$$\|\Delta A(t)x(t)\| \leq \|\Delta A(t)\| \|x(t)\| \leq \gamma \|x(t)\| \quad (69)$$

또한

$$\|\Delta A(t)\| \leq \gamma \quad (70)$$

이다. 식 (67)의 관계를 식 (70)에 적용하면

$$\|\Delta A(t)\| \leq \gamma < \frac{1}{2\lambda_{\max}(P)} \quad (71)$$

와 같으며 또는

$$\|\Delta A(t)\| < \frac{1}{2\lambda_{\max}(P)} \quad (72)$$

와 같이 된다. 식 (60)으로부터 최종적으로 섭동항에 대한 관계식은 다음과 같이 된다.

$$\|g(\Delta F(x,t))\| < \frac{1}{2\lambda_{\max}(P)} \quad (73)$$

이 결과로부터 마찰섭동의 norm은 행렬 P 의 최대 고유치와 반비례함을 알 수 있다. 예를 들어, 행렬 P 를 가장 간단한 positive definite 형태인 단위행렬로 정의하면 즉, $P=I$ 일 경우 $\lambda_{\max}(P)=1$ 이므로 식 (73)은 다음과 같다.

$$\|f(\Delta F(x,t))\| < \frac{1}{2} \quad (74)$$

식 (78)로부터 섭동시스템의 안정한 조건은 마찰 추정오차 함수의 norm이 항상 1/2보다 작아야 됨을 알 수 있다.

VIII. 컴퓨터 시뮬레이션

본 논문에서 제안한 제어알고리즘의 타당성을 검증하기 위하여 컴퓨터 시뮬레이션을 실시하였다. 그림 2의 시스템 모델에 사용된 파라미터값은 $m=10[kg]$, $I=5[kg \cdot m^2]$, $R=0.5[m]$, $r=0.05[m]$ 으로 설정하였으며, 마찰모델은 참고문헌 [11]에서 다룬 랜덤성분을 갖는 stick-slip 형태의 모델을 사용하였으며 수학적 모델은 다음과 같다.

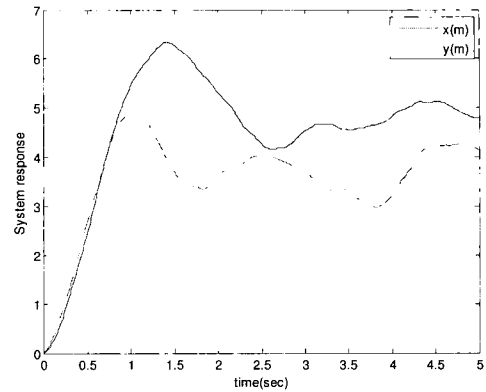
$$F = \left\{ d_0 + d_1 e^{-h_1|q|} + d_2 (1 - e^{-h_2|q|}) \right\} \text{sgn}(\dot{q}) \quad (75)$$

여기서 파라미터 d_0 , d_1 , d_2 , h_1 , h_2 는 uniform 확률분포를

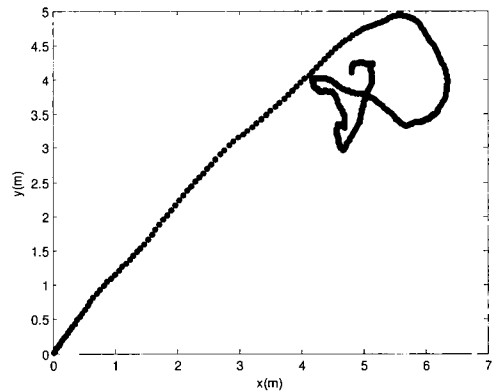
갖는 랜덤변수로 정의하였으며 그 범위는 $d_0=[0,1]$, $d_1=[0,2.2]$, $d_2=[0,1.8]$, $h_1=[0,1]$, $h_2=[0,0.0012]$ 와 같다.

시뮬레이션은 로봇의 위치제어와 궤적제어의 두 가지 시나리오에 대하여 실시되었으며 기존의 공칭제어기법 또한 실시하여 제안한 제어기법과 성능을 비교분석하였다. 로봇의 위치제어 시뮬레이션에 대하여 설정치는 $r_x=5$ 와 $r_y=4$ 로 하였으며 식 (6)의 공칭 제어파라미터 행렬은 참고문헌 [7]에서 제안한 제어기의 설계방법을 적용하여 산출하였으며 $K_1 = \text{diag}\{40,40,40\}$, $K_2 = \text{diag}\{10,5,4\}$ 와 같다.

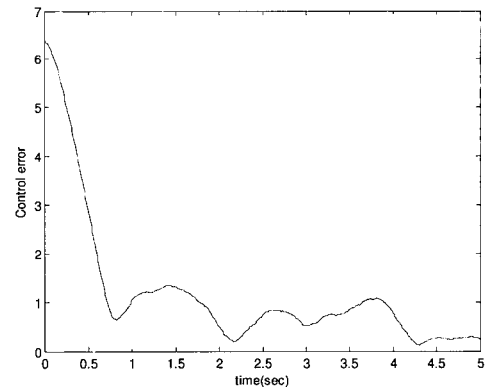
그림 4는 공칭제어시스템에 대한 시뮬레이션 결과파형을 보여준다. 그림 4(a)는 시스템 응답으로서 그림에서 알 수



(a) system response waveform.



(b) x-y trajectory.



(c) control error.

그림 4. 공칭제어시스템에 대한 시스템 응답.

Fig. 4. System response for the nominal control.

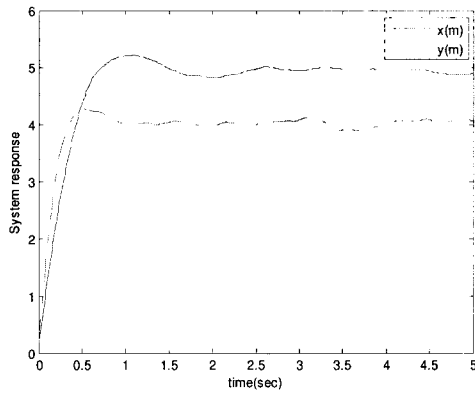
있듯이 주어진 제어시간동안 설정치에 도달하지 못하는 것을 알 수 있다. 공칭제어는 랜덤성분의 마찰력을 고려하지 않은 상태에서 설계가 이루어지며 이러한 현상에 대하여 바람직하지 못한 제어성능을 보여주고 있다. 그림 4(b)는 시스템의 두 위치응답 x 와 y 에 대한 궤적을 보여주고 있으며 어느 일정한 값에 수렴하지 못하는 것을 알 수가 있다. 한편 그림 4(c)는 제어오차 파형을 보여주며 오차방정식은 다음과 같이 정의하였다.

$$e(t) = \sqrt{(x_r(t) - x(t))^2 + (y_r(t) - y(t))^2} \quad (76)$$

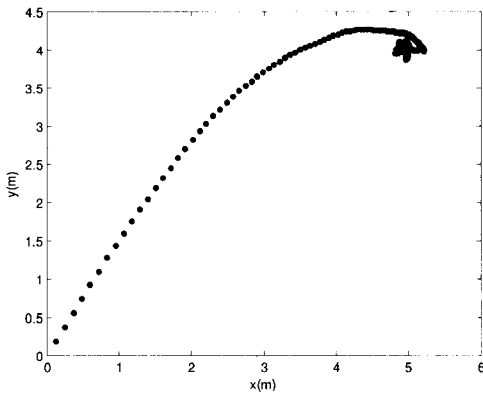
제어오차는 과도응답에서 최대값을 가지며 시간에 따라 감

소하지만 정상상태에서도 비교적 큰 오차범위를 가지며 오차신호를 통해서도 전제적으로 바람직하지 못한 제어성능을 볼 수가 있다.

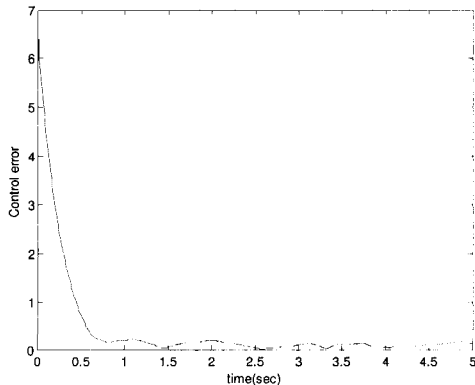
그림 5는 본 논문에서 제안한 제어기법을 적용하였을 때의 시스템응답을 보여준다. 그림 5(a)의 경우 초기의 과도응답에서 오버슈트가 발생하지만 그 후의 정상상태영역에서의 시스템 응답은 거의 설정치에 도달하는 것을 볼 수 있다. 정상상태에서의 약간의 리플현상이 발생하며 이러한 현상은 시스템의 랜덤마찰로 인한 것으로 매우 자연스러운 현상으로 분석되어진다. 그림 5(b)의 x 와 y 에 대한 위치궤적은 공칭제어시스템의 경우와는 달리 평형점에서 거의 수렴하는 것을 볼 수 있으며, 그림 5(c)의 제어오차 또한 과



(a) system response waveform.

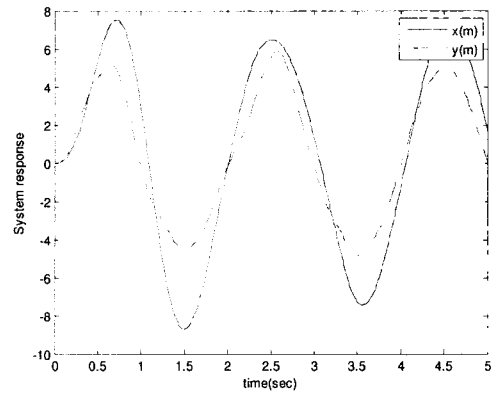


(b) $x - y$ trajectory.

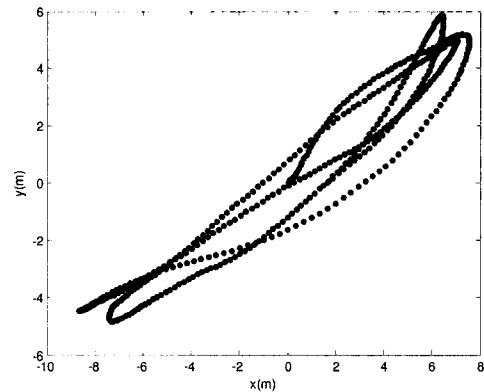


(c) control error.

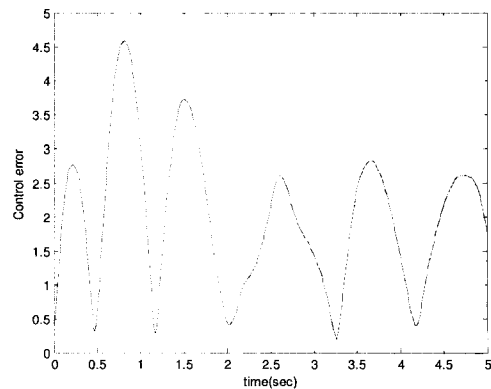
그림 5. 제안한 제어시스템에 대한 시스템 응답.
Fig. 5. System response for the proposed control.



(a) system response waveform.



(b) $x - y$ trajectory.

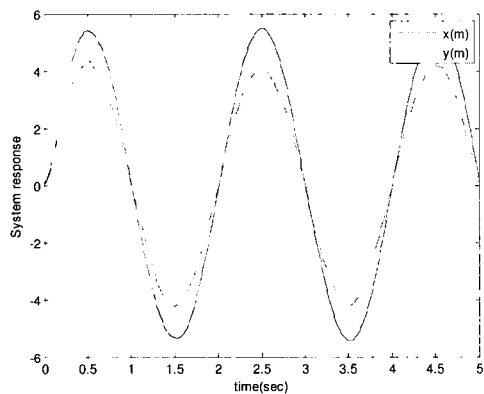


(c) control error.

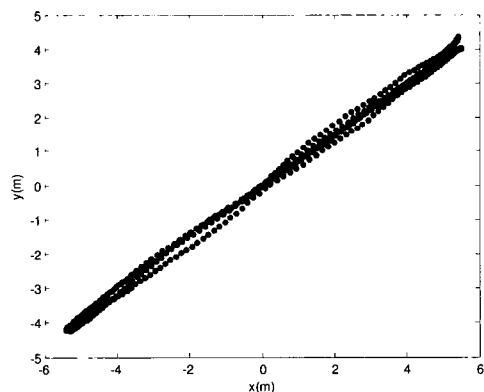
그림 6. 공칭제어시스템에 대한 시스템 응답.
Fig. 6. System response for the nominal control.

도응답에서 큰 오차가 발생되나 정상상태에서는 거의 0에 가까운 값을 보이고 있다. 그림 4와 5의 파형을 비교분석한 결과 제안한 제어기법의 성능이 현저히 개선되었음을 알 수 있다.

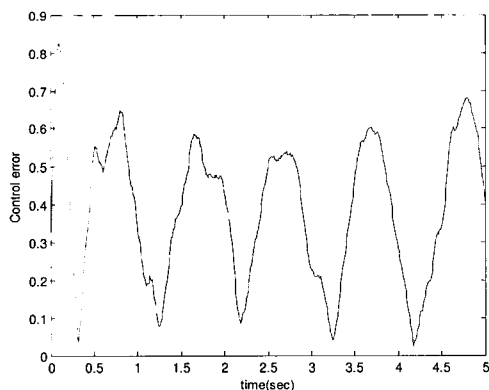
다음으로 시스템 궤적제어에 대하여 시뮬레이션을 실시하였으며 설정치는 정현파 형태로 하였으며 $r_x = 5\sin(\pi t)$ 와 $r_y = 4\sin(\pi t)$ 같이 정의하였다. 그림 6은 앞서 구현한 공칭제어시스템에 대한 시뮬레이션 결과파형을 보여준다. 그림 6(a)는 설정치에 대한 시스템의 두 위치에 대한 궤적 파형을 각각 보여주며 그림 6(c)에서 알 수 있듯이 최고오차가 4.5이며 평균오차가 약 1.8로 분석된다. 그림 6(b)는



(a) system response waveform.



(b) $x-y$ trajectory.



(c) control error.

그림 7. 제안한 제어시스템에 대한 시스템 응답.
Fig. 7. System response for the proposed control.

두 위치응답에 대한 $x-y$ 궤적을 보여주며 일정한 평형점에 수렴하지 못하는 것을 볼 수 있다.

그림 7은 제안한 제어기법에 대한 시뮬레이션 결과를 보여준다. 그림 7(c)의 제어오차를 통해 시스템 응답이 앞서 정의된 설정치에 만족스럽게 추종을 한다는 것을 볼 수 있으며 그림 7(b)에서 안정적인 시스템 궤적파형을 볼 수 있다. 그림 6과 7의 결과를 비교분석한 결과 제안한 제어시스템의 성능이 월등히 우수함을 입증할 수 있었다.

IX. 결론

본 논문은 마찰섭동을 갖는 자율이동 로봇시스템의 강인 적응제어기법을 제안하였다. 비선형 로봇시스템 모델은 공칭제어규칙을 산출하기 위하여 선형화기법을 통하여 근사화 되었으며, 시스템 섭동모델에 대한 강인제어를 구성하기 위하여 모델기반 제어기 설계법을 이용하였다. 보조제어규칙은 Lyapunov 안정성 이론을 만족하도록 설계되었으며 마찰력추정오차에 강인한 제어시스템으로 구성되었다. 컴퓨터 시뮬레이션을 통해 제어알고리즘의 타당성 및 성능의 우수성을 검증 및 분석하였다.

향후 연구로는 실제 자율이동 로봇시스템에 제안한 알고리즘을 적용하여 실시간 제어성능을 검증하여 산업현장에 적용 가능성 및 타당성을 분석한다. 또한 로봇시스템과 같은 복잡한 비선형성과 불확실성을 갖는 다양한 동적 시스템에도 본 제어알고리즘을 적용하여 그 응용범위를 확대하고자 한다.

참고문헌

- [1] D. Harter, "Evolving neurodynamic controllers for autonomous robots," *Proc. of Int. Joint Conf. on Neural Networks*, Canada, pp. 137-142, July 2005.
- [2] T. Ohnishi and T. Asakura, "Autonomous walking velocity control strategy for a spider-robot based on biological approach," *Proc. of the SICE Conf.*, Japan, pp. 382-387, Aug. 2005.
- [3] T. Chatchanayuenyong and M. Parnichkun, "Neural network based-time optimal sliding mode control for an autonomous underwater robot," *Mechatronics*, vol. 16, no. 8, pp. 471-478, 2006.
- [4] J. M. Ramirez, P. Gomez-Gil, and F. L. Larios, "A robot-vision system for autonomous vehicle navigation with fuzzy-logic control using lab-view," *Proc. of Electronics, Robotics & Automotive Mechanics Conf.*, pp. 295-300, 2007.
- [5] W. Ren, J.-S. Sun, R. Beard, and T. McLain, "Experimental validation of an autonomous control system on a mobile robot platform," *IET Control Theory & Applications*, vol. 1, no. 6, pp. 1621-1629, 2007.
- [6] F. N. Martins, W. C. Celeste, R. Carelli, M. Sarcinelli-Filho, and T. F. Bastos-Filho, "An adaptive dynamic controller for autonomous mobile robot trajectory tracking," *Control Engineering Practice*, vol. 16, no. 11,

pp. 1354-1363, 2008.

[7] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot using neural networks," *IEEE Trans. on Neural Networks*, vol. 9, no. 4, pp. 589-600, 1998.

[8] 조현철, 이권순, "확률마찰특성을 갖는 비선형 기계시스템을 위한 지능형 온라인 제어시스템," *대한전기학회*, vol. 56, no. 12, 2226-2232, 2007.

[9] J. M. Mendel, *Lessons in Estimation Theory for Signal*

Processing, Communications, and Control, Prentice Hall, 1995.

[10] H. K. Khalil, *Nonlinear Systems*, New Jersey, Prentice Hall, 2002.

[11] S.-W. Lee and J.-H Kim, "Robust adaptive stick-slip friction compensation," *IEEE Trans. on Industrial Electronics*, vol. 42, no. 5, pp. 474-479, 1995.



조 현 철

1971년 9월 1일생. 1997년 2월 부경대학교 제어계측공학과 졸업(공학사). 1999년 2월 동아대학교 전기공학과 졸업(공학석사). 2006년 8월 미국 네바다대학교 전기공학과 졸업(Ph.D.). 2006년~현재 동아대 전기공학과 제어 및

시스템 연구실 포닥연구원. 관심분야는 제어시스템, 고장진단, 확률 시스템, 신경회로망

이 권 순

제어·로봇·시스템학회 논문지 제14권 제8호 p. 784 참조.