

퍼지학습법을 이용한 크레인 시스템의 다변수 제어

임윤규*, 정병목**

Control for Multi-variable in Crane System using Fuzzy Learning Method

Yoon Kyu Lim*, and Byeong Mook Chung**

ABSTRACT

An active control for the swing of crane systems is very important for increasing the productivity. This article introduces the control for the position and the swing of a crane using the fuzzy learning method. Because the crane is a multi-variable system, learning is done to control both position and swing of the crane. Also the fuzzy control rules are separately acquired with the loading and unloading situation of the crane for more accurate control. The result of simulations shows that the crane is just controlled for a very large swing angle of 1 radian within nearly one cycle.

Key Words : Crane system(크레인 시스템), Fuzzy control(퍼지 제어), Learning algorithm(학습 알고리즘), Multi-variable system(다변수 시스템).

1. 서론

크레인은 산업 현장 전반에서 무거운 장비나 운반물을 옮길 때 많이 이용되고 있기 때문에 이 시스템에 대한 연구가 활발히 진행되고 있다⁽¹⁾. 이 시스템의 제어를 설계하기 위해서는 시스템의 전달함수(Transfer function)를 구해야 하는데 이를 구하기란 매우 힘든 경우가 많다. 또한 시스템의 마모나 주위 환경에 노출되어 있기 때문에 시스템의 변형은 막을 수 없게 된다. 이런 시스템의 변형이 생길 때마다 새로이 전달 함수를 새로 구해야 한다⁽²⁻³⁾. 이 두 가지 문제를 해결하기 위해서는 시스템의 전달함수 없이도 제어를 설계할 수 있어야 하고 시스템의 변형에 강건성을 가지거나 아니면 능동적으로 대처할 수 있는 제어가 필요하다⁽⁴⁻⁵⁾. 그래서 본 논문에서는 퍼지 학습

법을 사용하여 이 시스템의 제어를 설계하고자 한다. 그리고 크레인의 고유진동수는 고정된 것이 아니라 운반물을 잡고 놓는 과정에서 변하게 된다. 즉, 크레인의 고유진동수는 트레일러의 하중, 로프에 매달린 크레인 하중과 운반물의 하중의 합한 하중과 로프의 길이에 의해 결정된다. 만약, 로프의 길이가 일정하다면 트레일러의 하중과 크레인 및 운반물의 하중 비(ratio)에 의해 고유진동수가 결정된다. 이 비에 의해 크레인이 운반물을 잡기 전의 고유진동수와 잡고 난 후의 고유진동수가 다르며, 얼마만한 하중의 운반물을 잡느냐에 따라 달라진다는 것이다. 그리고 다변수 시스템이기 때문에 트레일러의 정확한 이송거리와 크레인의 진동을 동시에 제어하는 제어가 필요하다. 이 제어를 설계하는 것은 인간의 경험이나 전문가의 지식에 의해서는 어렵기⁽⁴⁻⁹⁾ 때문에, 본 논문에서는

* 영남대학교 대학원

** 영남대학교 기계공학부

퍼지 학습법을 쓰고자 한다. 사용한 퍼지 학습법은 기존의 성능지수표 대신에 원하는 진동각과 운반물의 실제 진동각 사이의 에러와 에러 변화량, 그리고 원하는 트레일러의 이동 거리와 실제 이동거리 사이의 에러와 에러 변화량을 포함하는 함수를 목적 함수로 정하여 이를 최소화하는 학습법이다⁽⁶⁻⁹⁾. 이 학습법을 이용하여 로프의 길이를 일정하게 두고, 운반물을 잡으러 갈 때의 퍼지 4차원 규칙 제어기와 운반물을 잡고 옮길 때의 퍼지 4차원 규칙 제어기를 설계하였다. 이 제어기를 LQR (Linear Quadratic Regulator)로 설계된 제어기와 비교하여 성능을 평가하였다.

2. 시스템의 고유값과 LQR 제어

2.1 시스템의 고유값

크레인 시스템은 Fig.1에서 보듯이 상단에 트레일러가 이동할 수 있는 궤도가 있다. 그리고 아래쪽으로 로프에 매달려서 운반물을 잡을 수 있는 크레인이 있고, 이 크레인 안에 담긴 운반물로 구성되어 있다. 트레일러가 이동할 때마다 로프에 매달린 크레인과 운반물의 흔들림이 발생하고, 크레인과 운반물의 흔들림에 의해 트레일러가 궤도 위에서 좌우로 흔들린다. Fig.1을 Lagrange 방정식을 이용하여 풀면 식(1)과 (2)가 된다.

$$(m_t + m_c)\ddot{x} + m_c l \cos \theta \ddot{\theta} - m_c l \sin \theta \dot{\theta}^2 = F \quad (1)$$

$$m_c l \cos \theta \ddot{x} + m_c l^2 \ddot{\theta} + m_c g l \sin \theta = 0 \quad (2)$$

여기서, m_t 는 트레일러의 하중, m_c 는 크레인과 운반물의 하중, l 는 로프의 길이, F 는 힘, 그리고 g 는 중력 가속도이다. 위 두 식(1), (2)를 \ddot{x} 과 $\ddot{\theta}$ 식으로 표현하면 식(3)과 (4)와 같다.

$$\ddot{x} = \frac{m_c \sin \theta (l \dot{\theta}^2 + g \cos \theta) + F}{m_c \sin^2 \theta + m_t} \quad (3)$$

$$\ddot{\theta} = - \frac{g \sin \theta (m_t + m_c) + m_c l \sin \theta \cos \theta \dot{\theta}^2 + \cos \theta F}{l (m_c \sin^2 \theta + m_t)} \quad (4)$$

위 식(3)은 트레일러의 이송에 대한 운동방정식이고, 식(4)는 크레인의 진동에 대한 운동 방정식이다.

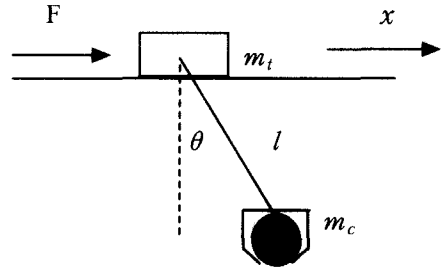


Fig. 1 Modeling for crane system.

이 시스템의 운동 방정식에는 댐퍼항이 없으므로 일반적인 매트릭스 형태로 표현하면 식(5)와 같다.

$$M \ddot{x} + Kx = 0 \quad (5)$$

여기서 M 은 질량 매트릭스이고, K 는 stiffness 매트릭스이다. 식(1)과 (2)를 선형화하고, 가진력이 없는 상태($F=0$)에 대해 식(5)와 같은 형태로 나타내면 식(6)과 같이 표현 할 수 있다.

$$\begin{bmatrix} m_t + m_c & m_c l \\ m_c l & m_c l^2 \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & m_c l g \end{bmatrix} \begin{bmatrix} x \\ \theta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (6)$$

위 식(6)을 이용하여 식(7)에서 고유값(eigen value) λ 를 구하기 위해서는 식(7)을 구하면 된다. 그 결과는 식(8)과 같다.

$$\det[K - \lambda M] = 0 \quad (7)$$

$$\lambda = \begin{bmatrix} -\frac{g}{l} \left(1 + \frac{m_c}{m_t}\right) & \\ 0 & \end{bmatrix} = \begin{bmatrix} -\frac{g}{l} (1 + \gamma) & \\ 0 & \end{bmatrix} \quad (8)$$

여기서, 크레인과 운반물의 하중과 트레일러의 하중 비(m_c/m_t)를 γ 로 정의하였다. 그리고 위 (8)식에 표현한 것처럼 2개의 모드(mode)가 나온다. 1차 모드는 rigid body 모드, 즉 m_t 와 m_c 가 같이 움직이는 모드를 의미하고, 2차 모드는 m_t 와 m_c 사이에서 상대적으로 움직이는 모드를 의미한다. 따라서 m_t 와 m_c 간의 진동은 로프의 길이 l 과 하중 비 γ 에 의해 좌우된다. 만약 로프의 길이를 일정하게 둔다고 가정하면, 크레인 시스템에 영향을 주는 진동은 γ 에 의해 좌우된다. 따라서 크레인이 운반물을 잡을 때와 잡지 않았을 때 m_t 와 m_c 의 진동 정도가 다르다는 것을 알 수 있다.

2.2 LQR 제어

LQR설계를 하기 위해서는 식(3)과 (4)를 상태 공간(state space)으로 나타낸 다음, Riccati 식을 이용하여 피드백 게인(feedback gain)을 구하려고 한다. 이를 위해 먼저, 일반적인 시스템의 상태 방정식은 식(9)로 표현 할 수 있다.

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}u(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) \end{aligned} \quad (9)$$

식(3)과 (4)를 상태공간인 식(9)로 표현하면, 시스템 행렬인 \mathbf{A} , \mathbf{B} , \mathbf{C} 는 각각 식(10)과 같이 된다.

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{m_c}{m_t} g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{(m_c + m_t)}{m_t l} g & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ \frac{1}{m_t} \\ 0 \\ -\frac{1}{m_t l} \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (10)$$

여기서, $\mathbf{x} = [x \ \dot{x} \ \theta \ \dot{\theta}]^T$ 이고 입력 $u = F$ 이다. LQR 제어기 설계에서 파라미터 변수는 상태 가중치 \mathbf{Q} 와 제어 가중치 R 이다. 이 두 값을 어떻게 정하느냐에 따라 제어기의 성능이 좌우된다. 본 논문에서는 운반물을 잡지 않은 상태, 즉 $\gamma = 2$ 일 때의 시스템 주파수는 식(8)식에 의해 0.61[hz]이므로 진동이 작기 때문에 상태 가중치 \mathbf{Q} 는 속도항(\dot{x} , $\dot{\theta}$)보다 상태항(x , θ)에 더 많은 가중치를 주어야 하므로 위해 식(11)과 같이 선정하였다.

$$\mathbf{Q} = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 5 \end{bmatrix}, \quad R = 0.0001 \quad (11)$$

위에서 선정된 \mathbf{Q} , R 을 LQR에서의 Riccati 식인 식(12)에 대입하면 식(13)과 같은 피드백 게인(\mathbf{K})을 구할 수 있다.

$$\mathbf{K}(t) \mathbf{A} + \mathbf{A}^T \mathbf{K}(t) + \mathbf{Q} - \mathbf{K}(t) \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{K}(t) = 0 \quad (12)$$

$$\mathbf{K} = [316.228, 489.396, 3.534, 270.578] \quad (13)$$

이 계인을 이용하여 제어한 결과를 Fig.4-5에 점선으로 나타내었다. 그리고 운반물을 잡은 경우, 즉 $\gamma = 10$ 일 때의 시스템 주파수 또한 식(8)에 의해 1.17[hz]이므로 진동이 $\gamma = 2$ 일 때 보다 크다. 그래서 상태 가중치 \mathbf{Q} 는 상태항(x , θ)보다 속도항(\dot{x} , $\dot{\theta}$)에 더 많은 가중치를 주어야 하기 때문에 식(14)와 같이 선정하였다.

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 50 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 50 \end{bmatrix}, \quad R = 0.0001 \quad (14)$$

위에서 선정된 \mathbf{Q} , R 을 앞에서와 마찬가지로 Riccati 식에 대입하면 식(15)와 같은 피드백 게인(\mathbf{K})을 구할 수 있다.

$$\mathbf{K} = [100.000, 850.007, -123.195, 78.509] \quad (15)$$

이 계인을 이용하여 제어한 결과를 Fig.6-7에 점선으로 나타내었다.

3. 퍼지 학습법

3.1 퍼지 제어

퍼지 논리 제어기란 프로세스의 입력과 출력의 애매한 관계를 인간의 경험과 지식을 기초로 한 언어적 제어규칙으로 표현하여, 이를 제어원리로 변환시킨 것이다. 일반적인 제어입력 $u(k)$ 는 다음과 같이 표현할 수 있다.

$$\begin{aligned} u(k) &= f[y_d(k+n), \dots, y_d(k+1), y(k), \dots, \\ & y(k-p+n), u(k-1), \dots, u(k-m+n)] \end{aligned} \quad (16)$$

여기서, n 은 시스템의 시간지연, m 은 입력 차수, p 는 시스템 차수, $y(k)$ 는 k 번째 출력, $y_d(k+n)$ 는 원하는 $(k+n)$ 번째 출력이다. $u(k)$ 는 과거의 입력과 출력뿐만 아니라 미래의 출력으로 이루어진 함수이다. 즉, 미래의 출력에 대한 정확한 예측이 가능해야만 현재의 제어 입력을 제대로 추정할 수 있다. 따라서 시스템의 역 모델을 구하기 위해 식(17)과 같이 퍼지 제어 입력을 에러와 에러 변화율들에 대한 함수로 나타낼 수 있다.

$$u(k) = F[e_i(k), c_i(k)], \quad (i = \theta, x) \quad (17)$$

여기서, $e_i(k) = y_d(k) - y_i(k)$,

$c_i(k) = [e_i(k) - e_i(k-1)] / \Delta t$ 이다. 퍼지 함의 R 은 식(18)과 같이 표현 할 수 있다.

$$R : E_{\theta}(e_{\theta}) \wedge C_{\theta}(c_{\theta}) \wedge E_x(e_x) \wedge C_x(c_x) \rightarrow v \quad (18)$$

여기서, 퍼지 후건부 값 v 는 계산의 용이성 때문에 비퍼지 값(defuzzified value)으로 사용하였다. 그리고 이를 k 번째 제어규칙 R_k 로 표현하면 식(19)와 같이 표현할 수 있다.

$$R_k : E_{\theta k}(e_{\theta k}) \wedge C_{\theta k}(c_{\theta k}) \wedge E_{xk}(e_{xk}) \wedge C_{xk}(c_{xk}) \rightarrow v_k \quad (k=1, 2, 3, \dots, M) \quad (19)$$

여기서, M 은 전체 규칙의 개수이다. 위 식을 멤버십 함수로 표현하면 식(20)과 같이 나타낼 수 있다.

$$R_k : \mu_{E_{\theta k}(e_{\theta k})} \wedge \mu_{C_{\theta k}(c_{\theta k})} \wedge \mu_{E_{xk}(e_{xk})} \wedge \mu_{C_{xk}(c_{xk})} \rightarrow v_k \quad (20)$$

위 식을 토대로 후건부의 값을 얼마나 반영해야 할 것인가를 결정하는 k 번째 규칙의 적합도(fitness) ϕ_k 는 식(21)과 같이 계산할 수 있다.

$$\phi_k = \frac{\mu_{E_{\theta k}(e_{\theta k})} \wedge \mu_{C_{\theta k}(c_{\theta k})} \wedge \mu_{E_{xk}(e_{xk})} \wedge \mu_{C_{xk}(c_{xk})}}{\sum_{k=1}^M \mu_{E_{\theta k}(e_{\theta k})} \wedge \mu_{C_{\theta k}(c_{\theta k})} \wedge \mu_{E_{xk}(e_{xk})} \wedge \mu_{C_{xk}(c_{xk})}} \quad (21)$$

제어 입력은 각각의 규칙에 대한 적합도와 비퍼지 값인 후건부의 곱을 합한 값이므로 식(22)와 같이 나타낼 수 있다.

$$u = \sum_{k=1}^M \phi_k \times v_k \quad (22)$$

3.2 학습 알고리즘

크레인 시스템을 제어하기 위한 제어 규칙의 학습은 모델규범형 퍼지 학습 제어를 사용하였다. 이 학습법은 원하는 출력과 시스템의 실제 출력의 차이를 구한 다음, 이를 퍼지 제어 규칙에 피드백하여 과거에 사용되었던 규칙을 수정하는 과정을 반복하여 제어를 설계하는 학습법이다.

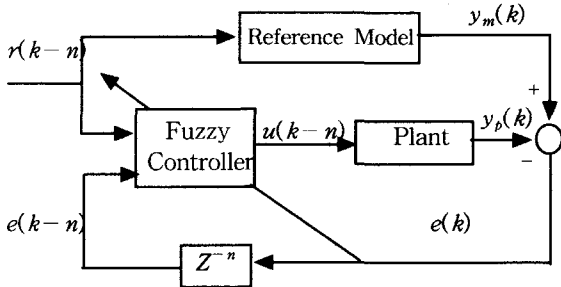


Fig. 2 Block diagram of the Model Based Fuzzy Control.

이 학습법에 대한 블록선도를 Fig.2에 나타내었다. 이 블록선도는 모델의 $(k-n)$ 번째 입력인 $r(k-n)$ 에 대해 모델의 k 번째 출력인 $y_m(k)$ 과 실제 시스템의 k 번째 출력인 $y_p(k)$ 의 차이인 $e(k)$ 를 구한다. 이를 n 스텝 시간지연을 시킨 $e(k-n)$ 를 학습에 반영하였다. 이 학습법을 사용하기 위해 먼저, 비용함수를 식(23)과 같이 에러와 에러변화율들의 함수로 설정하였다.

$$J(h) = \frac{w_{e_{\theta}}}{2} e_{\theta}(h)^2 + \frac{w_{c_{\theta}}}{2} c_{\theta}(h)^2 + \frac{w_{e_x}}{2} e_x(h)^2 + \frac{w_{c_x}}{2} c_x(h)^2 \quad (23)$$

여기서, $w_{e_{\theta}}$, w_{e_x} , $w_{c_{\theta}}$, w_{c_x} 는 가중치(weighting 상수)이다. 그리고 e_{θ} , c_{θ} 은 크레인의 진동각에 대한 에러, 에러 변화율이고, e_x , c_x 은 트레일러의 이송거리에 대한 에러, 에러 변화율이다. 위에서 정의한 비용함수를 최소화하기 위해 기울기법을 사용하여 식(24)와 같이 나타내었다.

$$\Delta v_i \propto - \frac{\partial (\sum_{k=0}^{\infty} J(k+h))}{\partial v_i} \quad (24)$$

현재의 입력이 n 스텝 지난 후의 k 번째 출력에 가장 큰 영향을 준다고 가정하여 이 출력에 대한 비용함수만 최소화하도록 퍼지 제어 규칙을 수정한다면 식(24)는 식(25)와 같이 표현할 수 있다.

$$\begin{aligned} \Delta v_i \propto & - \frac{\partial J(k+n)}{\partial u(k)} \frac{\partial u(k)}{\partial v_i(k)} \\ & = [w_{e_{\theta}} \frac{\partial y_{\theta}(k+n)}{\partial u(k)} e_{\theta}(k+n) \\ & + w_{c_{\theta}} \frac{\partial [y_{\theta}(k+n) - y_{\theta}(k+n-1)]}{\partial u(k) \Delta t} c_{\theta}(k+n) \\ & + w_{e_x} \frac{\partial y_x(k+n)}{\partial u(k)} e_x(k+n) \\ & + w_{c_x} \frac{\partial [y_x(k+n) - y_x(k+n-1)]}{\partial u(k) \Delta t} c_x(k+n) \frac{\partial u(k)}{\partial v_i} \end{aligned} \quad (25)$$

여기서, $\frac{\partial [y_i(k+n) - y_i(k+n-1)]}{\Delta t} = \dot{y}_i(k+n)$ 이고, $\partial u(k) / \partial v_i = \phi_i$ 이므로, 식(25)는 식(26)과 같이 표현된다.

$$\begin{aligned} \Delta v_i = & [w_{e_\theta} \frac{\partial y_\theta(k+n)}{\partial u(k)} e_\theta(k+n) \\ & + w_{c_\theta} \frac{\partial \dot{y}_\theta(k+n)}{\partial u(k)} c_\theta(k+n) \\ & + w_{e_x} \frac{\partial y_x(k+n)}{\partial u(k)} e_x(k+n) \\ & + w_{c_x} \frac{\partial \dot{y}_x(k+n)}{\partial u(k)} c_x(k+n)] \phi_i \end{aligned} \quad (26)$$

위 식(26)에서 자코비안(jacobian)항 만을 매트릭스로 나타내면 식(27)과 같다.

$$Jaco = \begin{bmatrix} \frac{\partial y_\theta(k+n)}{\partial u(k)} & \frac{\partial \dot{y}_\theta(k+n)}{\partial u(k)} & \frac{\partial y_x(k+n)}{\partial u(k)} & \frac{\partial \dot{y}_x(k+n)}{\partial u(k)} \end{bmatrix} \quad (27)$$

이 자코비안 매트릭스의 부호가 변하지 않는다면 부호가 같은 임의의 상수값을 사용해도 학습이 수렴한다는 것을 입증한 바가 있다^[10]. 본 논문의 크레인 시스템 경우도 자코비안의 부호가 바뀌지 않으므로 식(26)의 각각의 상수항과 자코비안의 곱을 학습율로 정의하여 식(28)과 같이 간단히 나타내었다.

$$\begin{aligned} \Delta v_i = & [\eta_{e_\theta} e_\theta(k+n) + \eta_{c_\theta} c_\theta(k+n) \\ & + \eta_{e_x} e_x(k+n) + \eta_{c_x} c_x(k+n)] \phi_i \end{aligned} \quad (28)$$

여기서, η_{e_θ} , η_{c_θ} 는 크레인의 진동각에 대한 에러, 에러 변화율의 학습율이고, η_{e_x} , η_{c_x} 는 트레일러의 이동거리에 대한 에러, 에러변화율의 학습율이다. 식(28)은 목적함수를 최소화하는 방향의 기울기를 나타내므로 규칙의 수정은 식(29)와 같이 나타낼 수 있다.

$$v_{i(new)} = v_{i(old)} + \Delta v_i \quad (29)$$

위 두 식(28)과 (29)를 이용하여 시뮬레이션 하였다.

4. 시뮬레이션

시뮬레이션 조건은 로프의 길이를 2m로 고정시키고, 초기 진동각을 57° (1 rad), 이동거리를 1m, 샘플링 시간은 0.01초로 하였다. 그리고 트레일러의 하중 m_i 는 100kg, 크레인의 하중을 200kg, 운반물의 하중을 800kg으로 하였다. 즉 운반물을 잡지 않았을 때 ($\gamma=2$) m_c 는 200kg, 운반물을 잡았을 때 ($\gamma=10$) m_c 는 1000kg으로 하여 실험하였

다. 퍼지 학습에 있어서 설계변수는 각각의 입력 변수에 대한 퍼지 집합의 수와 멤버십 함수의 형상, 학습지연시간 그리고 각각의 입력 변수에 대한 학습을 이렇게 4가지이다. 본 실험에서는 진동 각도에 대한 에러와 에러변화율 그리고 위치에 대한 에러와 에러 변화율을 입력변수로 하였다. 그리고 퍼지 집합의 수는 $5^4(5 \times 5 \times 5 \times 5)$ 으로 선정하였다. 그리고 멤버십 함수는 전 영역을 포함하는 강건성이 있는 가우스함수⁽⁷⁾를 사용하였으며, 퍼지 양자화(quantization)는 Fig.3에서와 같이 선정하였다.

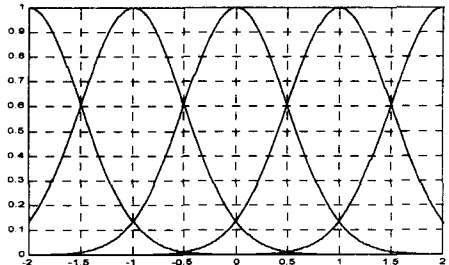


Fig. 3 Membership function for error and error change in crane and trailer.

그리고 학습지연 시간(n)은 2스텝으로 설정하였고, 학습율은 다음과 같이 설정하였다. 크레인 시스템은 다 변수 시스템이기 때문에 어느 한 쪽을 좋게 하면 다른 한쪽이 나빠지는 간섭(coupling)효과가 있다. 그래서 학습율을 각 변수의 에러와 에러 변화율에 대한 기여도를 동일하게 주어야 하지만, 트레일러의 과잉 응답(overshot)이 크게 나타나므로 이를 더 많이 고려하게되면 다른 요소의 응답은 다소 나빠지지만 전체 응답에서 볼 때 이익이 된다. 그래서 크레인의 진동각에 대한 에러 변화율을 학습에서 더 고려하여 학습율을 $\eta_{e_\theta}=16.7\%$, $\eta_{c_\theta}=16.7\%$, $\eta_{e_x}=16.7\%$, $\eta_{c_x}=50\%$ 로 선정하였다. 그 결과는 Fig.4-7에 실선으로 나타내었다. Fig.4와 5는 운반물을 잡지 않았을 경우에 대한 LQR과 퍼지 학습 결과이다. 위 결과에서 알 수 있듯이 LQR의 성능 보다 퍼지 학습한 결과가 다소 좋음을 알 수 있다. 이 때의 학습 횟수는 192회이다.

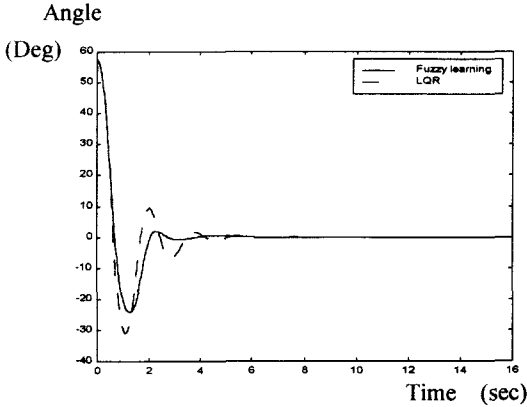


Fig. 4 LQR and fuzzy Learning response for swing of crane in $\gamma=2$.

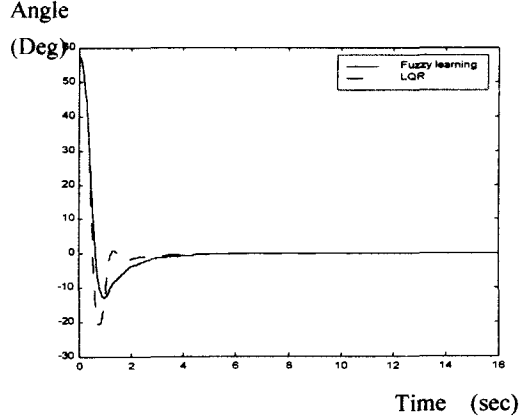


Fig. 6 LQR and fuzzy Learning response for swing of crane in $\gamma=10$.

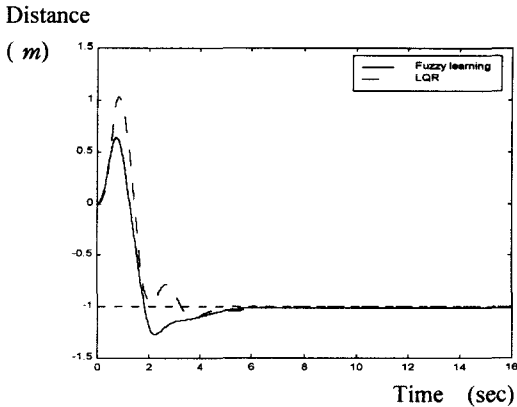


Fig. 5 LQR and fuzzy Learning response for distance of trailer in $\gamma=2$.

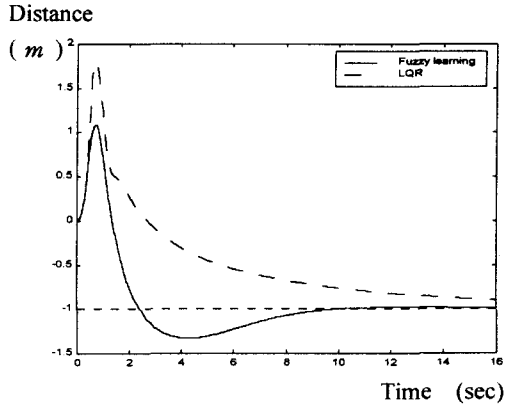


Fig. 7 LQR and fuzzy Learning response for distance of trailer in $\gamma=10$.

Fig.6과 7은 운반물을 잡았을 경우에 대한 LQR과 퍼지 학습 결과이다. Fig.7에서 보듯이 상승시간과 정착시간이 LQR 보다 퍼지 학습한 결과가 좋음을 알 수 있었다. 이 때의 학습 횟수는 307회이다. 이 학습에 소요된 시간은 펜티엄 PC(150MHz)에서 대략 $\gamma=2$ 인 경우는 2시간 정도 이고, $\gamma=10$ 인 경우는 3시간 정도 걸렸다. 그 학습된 결과 $\gamma=2$ 인 경우와 $\gamma=10$ 를 Fig.8과 9에 나타내었다. 이 그림은 4차원 규칙($e_\theta \times c_\theta \times e_x \times c_x$)의 전건부를 2차원 평면으로 하여 후건부 값(F)을 나타내었다. 그리고 4차원 규칙을 바로 나타낼 수 없기 때문에 e_θ 와 c_θ 에 대한 2차원 규

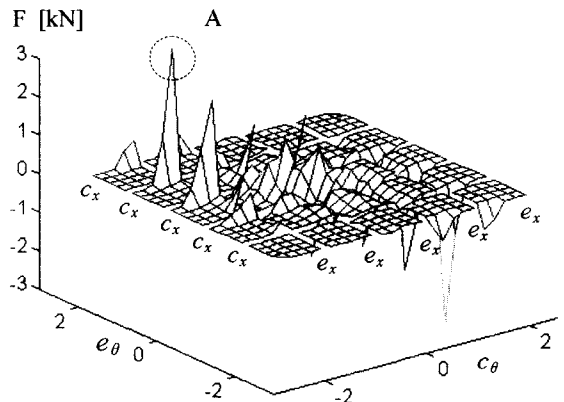


Fig. 8 Fuzzy rules after learning in $\gamma=2$.

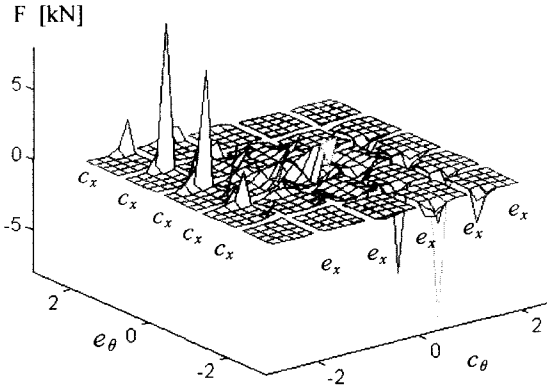


Fig. 9 Fuzzy rules after learning in $\gamma=10$.

칙 안에 e_x 와 c_x 에 대한 2차원 규칙을 5×5 개의 규칙 평면에서 나타낸 것이다. 그 예로 Fig.8의 점 A에 해당하는 규칙은 “if e_θ is positive and c_θ is negative big and e_x is positive big and c_x is positive big then 3.5 kN”이다.

5. 결론

본 논문에서는 크레인 시스템의 고유 진동수를 변화시키는 요인이 크레인을 지지하는 로프의 길이와 크레인 및 운반물의 하중과 트레일러의 하중 비임을 알 수 있었다. 그래서 크레인 시스템의 운행에 있어서 운반물을 잡고 놓는 행동을 반복하기 때문에 로프의 길이를 고정시키고 이 비가 달라지는 경우인 운반물을 잡은 경우와 잡지 않은 경우에 대해 각각의 제어기를 LQR과 퍼지 학습법을 이용하여 설계하였다. LQR 제어기 설계는 시스템의 전달 함수를 바탕으로 설계되기 때문에 퍼지 학습에 의해 설계된 제어기에 비해 좋은 성능을 가질 것이라 기대하였다. 그런데 결과는 반대로 퍼지 학습에 의해 설계된 제어기의 성능이 더 좋았음을 시뮬레이션을 통해 알 수 있었다. 그 이유는 선형화 과정에서 무시한 비선형항들 때문으로 즉, LQR 제어에서 사용하는 시스템의 전달 함수이며 초기 예러를 57도(1radian)의 큰 값을 주었기 때문이다. 따라서 다변수 시스템의 전달 함수를 구하기 힘든 경우 혹은 비선형성을 많이 가지고 있는 시스템의 경우는 본 논문에서 제안한

퍼지 학습법을 사용하는 것이 더 좋은 결과를 얻을 수 있음을 알 수 있었다.

참고문헌

1. 이호훈 and 조성근, "An anti-swing control for 2 axis overhead cranes," KACC, Vol. 5, No. 1, pp. 1428-1431, 1996.
2. W. Gharieb and G. Nagib, "Fuzzy control to multivariable systems case study : Helicopter Model," Fuzzy-IEE, Vol. 1, pp. 400-405, September, 1996.
3. M. A. Goodrich, W. C. Stirling and R. L. Frost, "A Satisficing Fuzzy Logic controller," Fuzzy-IEE, Vol. 1, pp. 272-276, September, 1996.
4. T. J. Procyk, "A Self-Organizing Controller For Dynamic Processes," Ph. D. Thesis, Queen Mary College, Univ. of London, 1977.
5. T. Procyk and E. Mandani, "A linguistic self-organizing process controller," *Automatica*, Vol. 15, No. 1, pp. 15-30, 1979.
6. S. Lee and R. Kil, "A Gaussian Potential Function Network with Hierarchically Self-Organizing Learning," *Neural network*, Vol. 4, pp. 207-224, 1991.
7. B. Chung and J. Oh, "Control of dynamic systems using fuzzy learning algorithm," *Fuzzy sets and Systems*, Vol. 59, No. 1, pp. 1-14, 1993.
8. B. Chung and J. Oh, "Autotuning method of membership function in a fuzzy learning control," *Journal of Intelligent & Fuzzy Systems*, Vol. 1, No. 4, pp. 335-349, 1993.
9. B. Chung and Y. Lim, "Hybrid Fuzzy Learning Controller for an Unstable Nonlinear System," *IATED*, Vol. 1, pp. 79-81, 1996.
10. B. Chung, "Control of nonlinear multivariable Systems using direct fuzzy learning method," *Journal of Intelligent & Fuzzy Systems*, Vol. 5, pp. 297-310, 1998.