

# CMAC에 의한 협동 퍼지 제어계의 운반차-막대 시스템 제어

## A Cooperative Fuzzy and CMAC Control for Cartpole System

권성규\*

Sunggyu Kwon

\* 계명대학교 기계자동차공학부

### 요 약

운반차-막대 시스템을 제어하기 위하여 두 개의 2 차원 퍼지 제어기가 CMAC에 의해 협동하게 하는 제어 전략을 개발하였다. 제어계에서 한 제어기는 운반차의 변위와 속도, 다른 제어기는 막대의 각도와 각속도를 각각의 2 개의 입력 변수로 하고 운반차에 가하는 힘이 두 제어기의 출력 변수인데, 이 변수를 외부의 감독에 따라 CMAC이 학습하게 하여 협동 제어의 효과를 발휘한다. 제어계 구성과 CMAC 훈련에 의한 협동 전략의 단순함에 비하여, 제어계는 4 개의 입력 변수에 의한 퍼지 제어기나 다른 해석적 방법에 의한 것에 비해 손색없는 제어 성능을 보였다.

### Abstract

A cartpole system is controlled by a control system consisting of two fuzzy controllers cooperating by a CMAC. Each controller uses 2 different input variables and yields the control force provided to the CMAC. The cooperation is due to training of the CMAC supervised by a judge which selects training information for the CMAC between two fuzzy controllers. The control scheme could be appreciated in terms of the tight structure of the controller, simple cooperating scheme due to the CMAC training, and accomplishing control goal that could not be attained by individual controllers.

**Key words** : CMAC, Cooperative Fuzzy Control, Cartpole System, Inverted Pendulum.

### 1. 서 론

그림 1은 어떤 운반차-막대 시스템(cartpole system[1])의 개략도이다. 이 시스템은 구조는 간단하지만 전형적으로 불안정한 비선형 시스템으로서 제어 기법을 개발하기 위해 실제의 비선형 문제를 연구하는 플랫폼으로 오랜 동안 이용되어 오고 있다.

이 시스템의 막대를 균형 잡는 일만을 고려하는 문제(도립진자(inverted pendulum) 제어)는 퍼지 기법을 소개하는 예제로도 흔히 사용된다. 운반차의 위치는 고려하지 않고, 단지 막대의 균형 잡기만을 다루기 때문에, 막대의 기운 각도와 그 각속도가 퍼지 제어기의 2 개의 입력변수로 사용된다 [2]. [3]에서는 막대 균형 잡기 문제에 대해서 막대의 각도와 그 각속도에 따라 시스템 상태 공간을 5 영역으로 구분하고, 유전자 알고리즘으로 각 영역에 적절한 제어 게인 값을 할당하는 퍼지 제어 기법을 제안하였다.

막대의 각도 제어에 더하여, 운반차의 위치도 제어하는 문제에 적용하는 제어 기법도 많이 개발 되어 있다. 이를 위해서 퍼지 제어에서는 보통 운반차-막대 시스템의 상태를 정의하는 4 개의 변수가 모두 입력 변수로 사용된다. 그러면, 제어기 규모가 커지고 많은 추론 규칙이 필요하게 되며, 그 많은 규칙들을 체계적으로 설정하기가 쉽지 않다. [4]에서는 4 개의 입력 변수로 인한 108 개의 제어 규칙을 설정하였는

데, 이런 경우에 제어 규칙을 정하기 위해서 유전자 알고리즘과 같은 탐색 알고리즘을 이용하기도 하지만 여전히 버거운 작업이다. 또한, 제어 규칙이 퍼지 제어기의 성능에 미치는 영향이 크기 때문에, 많은 제어 규칙을 정해야 하는 제어 기법은 좋지가 않다.

운반차-막대 시스템의 4 개의 입력 변수에 의한 제어 규칙 설정의 어려움을 피하는 대신, 시스템의 제어 특성을 제어기 설계에 반영하는 다양한 기법들이 개발되고 있다. [5]에서는 두 세트의 퍼지 규칙 모듈로 구성되는 퍼지 제어기를 구축하였는데, 한 모듈은 제어된 변수의 크기를 위한 것이고, 다른 모듈은 그 변수의 부호를 위한 것이다. 운반차의 위치를 막대의 각도에 영향을 미치는 외란으로 간주하고, 운반차의 위치를 막대의 가상 각도 목표로 변화하여, 그 가상의 각도가 막대의 각도 제어에 반영되도록 하였다. [6]에서는 막대가 트랙의 중앙으로 향하여 기울어지도록 하는 일을 막대의 각도 제어 문제로, 운반차를 트랙의 중앙으로 움직이게 하는 일을 운반차 위치 제어 문제로 설정하여, 막대의 각도 제어를 운반차의 위치 제어보다 더 우선으로(priority order) 처리하는 제어 전략을 개발하였다. [7]에서는, 운반차-막대 시스템의 제어 조건들을, 운반차가 트랙의 중앙으로부터 벗어나 있거나 막대가 기울어져 있는 출발 모드('departure mode')와 막대의 균형을 잡으면서 운반차를 트랙의 중앙으로 움직일 수 있는 접근 모드('approaching mode'), 두 유형으로 구분하였다.

이런 연구 결과를 분석하면, 막대의 각도 제어를 위해서는, 막대의 각도와 그 각속도를 2 개의 입력 변수로 사용하

접수일자 : 2006년 1월 6일

완료일자 : 2006년 6월 5일

는 것이 분명하다. 그러나 운반차의 위치 제어를 위해서, 예를 들어, 운반차의 위치와 그 선속도를 2 개의 입력 변수로 퍼지 제어기에 사용한 연구 예는 찾아보기 힘들다.

운반차-막대 시스템에 대해서 고전적인 제어기와 퍼지 제어를 혼합하여 사용하는 예도 있다. [8]에서는 운반차-막대 시스템의 swing-up 제어를 위해, 퍼지 제어기와 선형 상태 피드백 제어를 결합하는 혼합 제어 기법을 개발하였다. 또한, PID 제어기와 퍼지 제어를 결합하는 조합 제어 기법도 있다 [9].

한편으로, CMAC과 퍼지 제어를 혼합하는 제어 전략도 흔하다. CMAC은 제어기가 담당할 제어 함수를 훈련을 통해 순람표(look-up table)에 넣고, 플랜트의 상태에 해당하는 순람표의 값을 끄집어내어 제어기의 출력 값으로 사용한다[15]. [10]에서는 CMAC의 원래 중간 변수 변환 함수들을 퍼지 이론의 소속 함수들로 대체하여 CMAC의 출력을 부드럽게 하고 함수 어둠 능력을 향상하였다. [11]에서는 강화 학습을 가속하기 위하여 CMAC의 구조를 개조하고, 운반차-막대 시스템의 막대 균형 잡기를 위한 퍼지 제어기의 출력이 개조된 CMAC 구조에 통합되게 하였다.

운반차-막대 시스템이 아닌 다른 문제들에 대해서, 두 개의 퍼지 제어기를 혼합 사용하는 예도 있다. [12]에서는 교통의 원활한 소통을 위한 교통 신호 제어 시스템을 개발하기 위해서 두 개의 퍼지 제어기가 협동하는 제어 방법을 개발하였다. 두 퍼지 제어기 중, 첫 번째 수준의 제어기는 교통 상황을 나타내는 데이터로부터 소통의 긴급 정도를 나타내는 출력('urgent degree')을 생산하고, 두 번째 수준의 퍼지 제어기는 그 출력을 입력으로 받아들여 교통 신호 체계의 변화를 유발하는 출력을 내 보낸다. [13]에서는 대형 오일 전송 시스템의 특성에 따라, 두 개의 퍼지 제어기로 구성되는 제어 전략을 개발하였는데, 제어계의 제어 변수는 두 퍼지 제어기의 출력 변수 값을 혼합(blending)해서 생성하였다. 입력 변수는 2 개로, 한 퍼지 제어기에 대해서는 주 펌프의 흡입 압력과 파이프 유량이고, 다른 퍼지 제어기에 대해서는 주 펌프의 흡입 압력과 펌프 토출 압력이며, 출력 변수는 두 퍼지 제어기에 대해서 공히 펌프의 회전 속도의 변화이다. 펌프의 회전 속도 변화를 나타내는 두 수치 값을 공식에 따라 혼합하여 제어계의 출력 값을 생성하였다.

이 논문에서는 운반차-막대 시스템의 막대의 각도 제어와 운반차의 위치 제어를 위해 2 개의 입력 변수로 구성되는 퍼지 제어기 두 기를 설계하여, 외부 감독자의 판단에 따라 두 퍼지 제어기의 출력을 학습하는 CMAC이 운반차-막대 시스템에 힘을 작용하게 하는 제어계를 개발하고, CMAC에 의한 두 퍼지 제어기의 협동 특성과 막대의 각도 제어와 운반차 위치 제어가 구현되는 성능을 조사하였다.

서론에 이어, 2 절에서는 운반차-막대 시스템의 제어에 관한 기초 사항을, 3 절에서는 2 개의 입력 변수들로 구축되는 두 개의 퍼지 제어기와 그 둘이 CMAC에 의해 협동하는 제어계에 대해 설명한다. 4 절에서는 두 퍼지 제어기의 협동을 구현하는 CMAC의 훈련에 관해 살핀 후, 5 절에서 이 논문에서 개발한 제어계의 특성에 대해 토의하고 결론을 맺는다.

## 2. 운반차-막대 시스템 제어

그림 1에서,  $x$ 는 운반차의 변위,  $\dot{x}$ 는 운반차의 선속도,  $\phi$ 는 막대의 기운 각도,  $\dot{\phi}$ 는 그 각속도이다. 운반차는 직선 트랙을 따라서 자유로이 움직일 수 있고 그 운반차에 실려서 있는 막대는 운반차와 트랙을 포함하는 수직 평면에서 자유

로이 회전할 수 있다. 운반차-막대 시스템의 지배 운동방정식은 다음과 같다: [14]

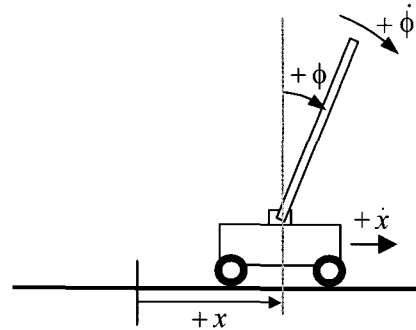


그림 1. 운반차-막대 시스템의 개략도.

Fig. 1. The schematic diagram of the cartpole system.

$$\ddot{\phi} = \frac{g \sin\phi - \cos\phi \left[ \frac{F + m l \dot{\phi}^2 \sin\phi - \mu_c \operatorname{sgn}(\dot{x})}{m_c + m} \right] - \frac{\mu_p \dot{\phi}}{m l}}{l \left[ \frac{4}{3} - \frac{m \cos^2\phi}{m_c + m} \right]}$$

$$\ddot{x} = \frac{F + m l [\dot{\phi}^2 \sin\phi - \ddot{\phi} \cos\phi] - \mu_c \operatorname{sgn}(\dot{x})}{m_c + m}$$

여기서, 중력 가속도  $g = -9.8 \text{ m/sec}^2$ , 운반차 질량  $m_c = 1.0 \text{ kg}$ , 막대의 질량  $m = 0.1 \text{ kg}$ , 막대 길이의 반  $l = 0.5$ , 트랙에서 운반차의 마찰계수  $\mu_c = 0.0005$ , 운반차에서 막대의 마찰계수  $\mu_p = 0.000002$ , 운반차의 질량 중심에 가하는 힘  $F$  Newtons.

제어 목표는 시스템이 임의의 초기 상태에서부터 해제 되었을 때, 제어기는 운반차를 트랙의 중앙에 위치시키되 그러한 동안 운반차에 실린 막대는 넘어지지 않고 균형을 유지하도록 하는 것이다. 이를 위해, 제어기는 매 제어 시각마다 운반차 중심에 힘을 가한다. 막대에는 아무 힘도 직접 작용하지 않아서, 막대의 운동은 오로지 운반차에 가해지는 힘에 의해 변화된다.

막대의 균형을 유지하면서 운반차를 트랙 중앙으로 옮기는 제어 문제는 예를 들어, 2 쪽 보행과 수직 발사 때 로켓의 안정화 등과 같은 균형 잡기 상황에서 나타나는 본래 불안정하고, 여러 개의 출력을 가진 동적 시스템의 예로서 흔히 사용 된다 [4].

우리가 손바닥에 긴 막대를 거꾸로 세우고 막대의 균형 잡는 실험을 한다고 하면, 손이 그림 1의 운반차에 해당한다. 직관과 경험에 의하면, 이 시스템을 제어하는 일은 다음과 같이 요약할 수 있다. 만약 막대가 오른쪽으로 기울면( $+\phi$ ) 팔에 오른쪽으로 힘을 주어 손을 오른쪽( $+x$ )으로 재빨리 움직여서 막대가 거의 왼쪽으로 기울( $-\phi$ ) 정도로 해서 손은 왼쪽으로 천천히 움직여( $-\dot{x}$ ) 막대의 균형을 잡으면서 몸 중심(트랙의 원점)으로 손을 옮겨올 것이다. 반대로, 만약 막대가 왼쪽으로 기운다고 하면, 이번에는 팔을 왼쪽으로 힘을 주어 손을 왼쪽으로 재빨리 움직여서 막대가 거의 오른쪽으로 기울어지게 한 다음, 손을 오른쪽으로 천천히 움직여 막대의 균형을 잡으면서 손을 몸 중심으로 옮겨올 것이다. 이것은 몇 번 만 해보면 쉽게 할 수 있는 일이다.

이것을 상기하면, 서론에서 살펴 본, 연구 [6]에서의 동적

중요도('dynamic importance')나 [7]에서 출발 모드와 접근 모드가 고안된 배경을 이해할 수 있다.

### 3. CMAC에 의한 협동 퍼지 제어

기존 연구들에 나타난 바와 같이, 운반차-막대 시스템을 운반차의 위치와 함께 막대의 각도도 제어하기 위해서는 시스템을 정의하는 4 개의 상태 변수 값이 어떤 형태로든지 퍼지 제어기에 반영되어야 한다. 물론, 4 개의 상태 변수를 모두 퍼지 제어기의 입력 변수로 사용하면 제어기의 차원이 증대되는 문제와 비현실적으로 많은 수의 제어 규칙을 설정해야 하는 어려움을 맞게 된다.

먼저, 이 논문에서는, 운반차의 변위  $x$ 와 그 속도  $\dot{x}$ 를 2 개의 입력 변수로 사용하는 퍼지 제어기( $x$ -제어기)가 막대의 균형도 잡을 수 있을 뿐만 아니라 운반차의 위치도 제어할 수 있는 예를 보이고자 한다. 사실, 도립진자의 균형 잡기를 위한 퍼지 제어를 포함한 다양한 연구에서, 운반차의 변위와 속도를 2 개의 상태 변수로 이용하는 제어기를 본 적이 없다.

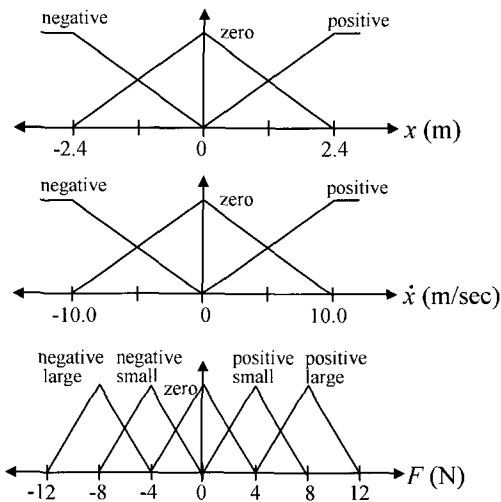


그림 2.  $x$ -제어기를 위한 2 개의 입력 및 1 개의 출력 변수들에 대한 소속 함수.

Fig. 2. Membership functions for 2 inputs and 1 output variable for the  $x$ -fuzzy controller.

표 1. 2 개의 입력 변수  $x$ 와  $\dot{x}$ 에 대한 추론 규칙 표.

Table I. Rule table with 9 rules for 2 input variables,  $x$  and  $\dot{x}$ .

F		$\dot{x}$		
		N	Z	P
$x$	N	PL	PS	Z
	Z	PS	Z	NS
	P	Z	NS	NL

이  $x$ -제어기를 위해서, 2 개의 입력 변수( $x$ 와  $\dot{x}$ )와 1 개의 출력 변수( $F$ )에 대한 소속(membership) 함수는 그림 2에서 보는바와 같다. 퍼지 로직이 제시한 입력들로부터 적절한

출력을 만들어낼 추론 규칙은 표 1과 같다. 이 추론 규칙은 2 절에서 살펴본 제어 전략에 의하면 쉽게 만들어 낼 수 있는 것이다. 표 1에서 Z는 그림 2에서 zero를, 예를 들어, NS는 negative small, PL은 positive large 등을 각각 의미한다. 그리고 F는 출력 변수로서, 운반차에 가하는 힘을 나타낸다.

제어 시뮬레이션을 위해서, 2 절에서 제시한 운반차-막대 시스템 운동방정식에 대해 Runge-Kutta 4th-order 방식으로 수치해를 구하였고,  $\Delta t = 0.02$ 이다. 시스템의 초기 상태가 (1.0, 0.4, -9.5, -56.0)일 때,  $x$ -제어기에 의한 시스템의 행태는 그림 3에서 볼 수 있다.

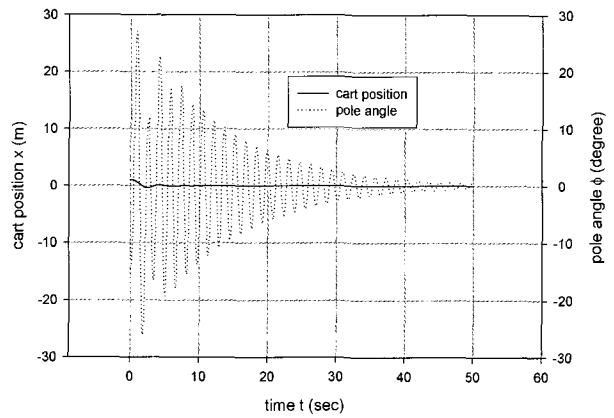


그림 3.  $x$ -제어기에 의한 운반차의 위치와 막대의 각도 변화.

Fig. 3. Cart position and pole angle versus time for the  $x$ -fuzzy controller.

그림 3에 의하면 초기 상태에서 해제된 운반차는 금방 트랙의 중앙으로 이동하는 반면, 막대가 50 초가 되도록 지속적으로 요동하고 있음을 알 수 있다. 시스템의 이런 행태는 다른 초기 조건들에 대해서도 유사하다. 이 시스템 행태를 관찰하면  $x$ -제어기가 중요한 제어 역할을 가지고 있음을 알 수 있다.

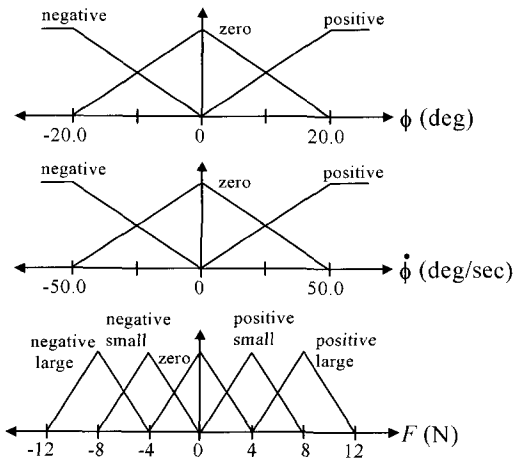


그림 4.  $\phi$ -제어기를 위한 2 개의 입력 및 1 개의 출력 변수들에 대한 소속 함수.

Fig. 4. Membership functions for 2 inputs and 1 output variable for the  $\phi$ -fuzzy controller.

표 2. 2 개의 입력 변수  $\phi$  와  $\dot{\phi}$  에 대한 추론 규칙 표.  
Table 2. Rule table with 9 rules for 2 input variables,  $\phi$  and  $\dot{\phi}$ .

F		$\phi$		
		N	Z	P
$\dot{\phi}$	N	NL	NS	Z
	Z	NS	Z	PS
	P	Z	PS	PL

막대가 똑바로 서기까지 비록 시간이 오래 걸리기는 하지만, 궁극적으로 시스템은 제어 목적에 부합하는 상태에 이르게 된다. 다시 말하면, 운반차의 변위  $x$ 와 그 속도  $\dot{x}$ 를 2 개의 입력 변수로 가지는 퍼지 제어기는 막대가 균형을 잡기 훨씬 전에 운반차를 트랙의 중앙에 위치시킨다. 단적으로, 이  $x$ -제어기는 막대가 넘어지지 않게 하면서 운반차를 확실하게 트랙 중앙으로 몰아간다.

다음으로, 운반차-막대 시스템의 막대 각도 제어(도립진자의 균형 잡기)를 위한 통상적인 퍼지 제어기를 구축하기 위해, 막대의 각도  $\phi$ 와 그 속도  $\dot{\phi}$ 를 2 개의 입력 변수로 사용하고, 운반차에 가하는 힘을 출력 변수로 사용한다. 이  $\phi$ -제어기를 위한 2 개의 입력 변수  $\phi$ 와  $\dot{\phi}$  및 출력 변수( $F$ )에 대한 소속 함수는 그림 4와 같고, 출력을 위한 추론 규칙은 표 2와 같다. 이  $\phi$ -제어기에 의한 운반차-막대 시스템의 행태는 그림 5에서 볼 수 있다.

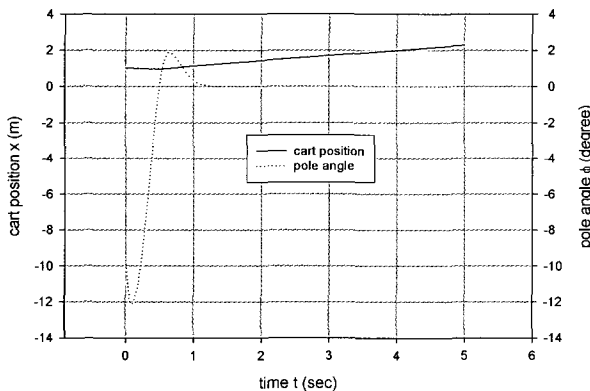


그림 5.  $\phi$ -제어기에 의한 운반차의 위치와 막대의 각도 변화.

Fig. 5. Cart position and pole angle versus time by the  $\phi$ -fuzzy controller.

그림 5에 의하면,  $\phi$ -제어기가 거의 1 초 만에 막대의 균형을 잡는 것을 알 수 있다. 이것은 그림 3의 시스템 행태와 현저하게 대조되는 특징이다. 비록 운반차는 계속 트랙 중앙으로부터 멀어지고 있어서 운반차의 변위는 도저히 제어할 수 없지만, 막대 균형 잡기는 틀림없이 달성되는 것을 알 수 있다. 이것은 도립진자의 균형 잡기 문제에서 볼 수 있는 전형적인 시스템의 행태이다.

그러면, 막대 각도 제어를 보장하는  $\phi$ -제어기와, 운반차 위치 제어를 보장하는  $x$ -제어기를 [7]에서처럼, 시스템의 상태에 따라 적절히 선택적으로 사용함으로써 막대의 각도 제

어와 운반차의 위치 제어를 동시에 달성할 수 있을 것으로 기대할 수 있다. 그러므로 그림 3과 그림 5에서 보는 두 퍼지 제어기의 바람직한 특성이 조합되어 시스템에 구현되게 하는 ‘협동’ 제어 전략이 필요하다. 문제는 두 퍼지 제어기를 어떻게 협동하게 할 것인지에 관한 전략과 ‘순간순간 어느 제어기의 출력을 플랜트로 내 보낼 것인가?’ 하는 것이다. 이 문제와 관련하여, [8]에서는 운반차-막대 시스템의 swing-up 제어를 위해서, 퍼지 제어와 선형 상태 피드백 제어 사이에서, 퍼지 제어로부터 선형 피드백 제어로의 전환하는 때를 정하기 위해 state-matching 조건들을 설정하고 그것들을 저장하였다가 시스템 상태가 저장된 조건과 부합하면 그때 제어기를 전환하였다.

이제, 이 논문에서 개발한 CMAC에 바탕을 둔 두 퍼지 제어기의 협동 제어 전략을 그림 6으로 설명한다. 그림에서 서로 다른 2 개의 입력 변수로 정의되는 두 퍼지 제어기는 4 개의 입력 변수로 정의된 CMAC을 위한 학습 정보를 제공하고, judge(평가자)는 특정 제어 시각에 CMAC이 학습을 할 것인지 말 것인지를 결정하고, 학습을 해야 한다면 두 퍼지 제어기 중 어느 제어기의 출력을 CMAC의 학습 정보로 인가할 것인지를 결정한다. CMAC은 평가자의 감독에 따라 제어 정보를 학습하여 순람표에 저장하고 플랜트 제어를 위한 출력을 내 보낸다. 스위치는 CMAC의 학습이 없는 동안은 중립 위치에 머문다. 그래서 스위치에는 3 개의 단자가 있다.

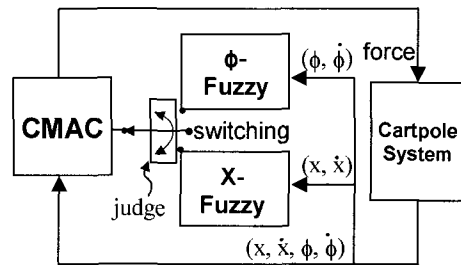


그림 6. CMAC에 의한 두 퍼지 제어기의 협동 제어계의 개략도.

Fig. 6. The diagram of the control scheme of cooperative fuzzy control due to CMAC.

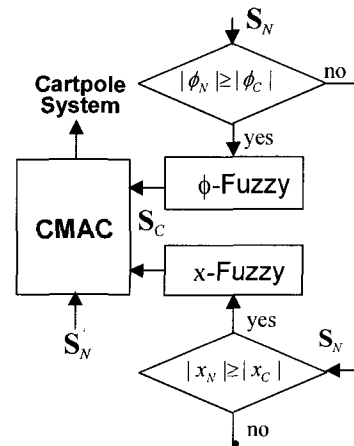


그림 7. CMAC에 의한 퍼지 제어기의 협동 제어를 위한 스위칭 알고리즘.

Fig. 7. The switching algorithm for the cooperative fuzzy control

평가자가 CMAC을 위한 학습 정보를 제공하기 위하여 스위치를 설정하는 알고리즘은 다음과 같다: 이를 위해 그림 7에서,  $s_c$ 는 현재의 운반차-막대 시스템 상태,  $s_N$ 은 CMAC에 의해 플랜트에 제어 힘이 작용한 뒤에 달라진 시스템의 상태를 나타낸다. 기본적으로, 그림 6에서 평가자는,  $s_N$ 에 대해서 CMAC이 출력 값을 플랜트에 작용하기 전에, 필요하다면 CMAC이  $s_c$ 에 대한 훈련을 마치도록 하여야 한다. 최초의  $s_c$ 에 대해서 CMAC은  $\phi$ -제어기로부터의 출력 값을 순람표에 저장함과 동시에 그 값을 운반차에 작용한다. 평가자는,  $s_N$ 에 대해, 막대의 기운 각도를 우선적으로 관찰한다. CMAC이 플랜트에 바로 전에 인가한 힘에 의해서, 막대의 기울기가 작아지지 않아서, 상황이  $|\phi_N| \geq |\phi_C|$  되면, CMAC에 의한 출력이 바람직하지 않다고 판단하고, CMAC이  $s_c$ 에 대한  $\phi$ -제어기의 출력을 학습하게 한다. 그러기 위해 스위치는  $\phi$ -제어기로 연결된다. 그러나, 막대의 기운 각도가 더 커지지 않았다면, 다음으로, 운반차가 트랙 중앙으로부터 더 멀어졌는지를 관찰한다. 운반차가 트랙 중앙으로부터 더 가까워지지 않아서,  $|x_N| \geq |x_C|$  이면, CMAC이  $x$ -제어기의 출력을 학습하도록 스위치를 설정한다. 그래서,  $|\phi_N| \geq |\phi_C|$  이면서  $|x_N| \geq |x_C|$  인 상황에 대해서는 CMAC은 우선적으로  $\phi$ -제어기의 출력을 학습하게 된다. 상황이  $|\phi_N| < |\phi_C|$  이고  $|x_N| < |x_C|$  이면,  $s_c$ 에 대해 CMAC은 학습을 하지 않는다. 어느 경우이건 CMAC이 학습 정보를 접수하면 스위치는 중립 위치로 복귀한다.

그림 7에서, 스위치의 설정을 위해  $s_N$ 이 제공되지만, CMAC은  $s_c$ 에 대해서 학습하는 것을 주목한다. 즉, CMAC은  $s_c$ 에 대한 퍼지 제어기의 출력을 학습을 하지만, 운반차-막대 시스템에 내놓는 CMAC의 출력은 스위치 설정을 위해 제공된  $s_N$ 에 대한 것이다.

#### 4. CMAC의 훈련

CMAC을 구축하기 위해서 예를 들어,  $K=5$ 로 하고,  $x$  축에 대해서는 8 개, 나머지 변수 축들에 대해서는 10 개의 구간으로 양자화하여, 각 중간 변수에 대해서 8,000 개의 메모리 셀을 할당하였다. 막대를 똑바로 세우면서 운반차를 트랙 중앙으로 몰아가야 하는 제어 목표를 고려하여,  $\phi$  축과  $x$  축에 대해서 변수 값 0 주위로 더 조밀하게 불균일하게 양자화하였다 [15]. 특히,  $x$  축은 운반차-막대 시스템의 대칭적인 제어 특성[16]을 반영하여 양의 축만 입력 변수 축으로 설정하였다. 그로인해 메모리 규모를 축소하고 CMAC의 학습에서 시스템 제어의 대칭성을 구현하였다.

애초에 CMAC은 전혀 훈련이 안된 상태이다. 제어 이력이 누적됨에 따라, CMAC의 출력이 적절하지 않다고 평가자가 판단한 경우에 두 퍼지 제어기 중의 어느 하나의 출력을 CMAC이 학습하며 제어계의 역량을 확립해 간다. CMAC의 훈련은 평가자의 판단에 따르고, 훈련 모델은 두 퍼지 제어기이다. CMAC은 플랜트의 상태를 나타내는 4 개의 변수에 의해 정의되어 있지만, 퍼지 제어기는 단지 2 개의 변수들로 정의되어 있다. 그래서, 사실 퍼지 제어기가 CMAC에 제공하는 훈련 정보는 플랜트의 상태를 나타내는 것도 아니고 제어계의 제어 목적에 비추어 온전하지도 않다. 그러므로, CMAC은 온전하지 않은 훈련 정보를 학습하여 플랜트를 제어하기에 적절한 출력 정보를 생산하는 역할을 한다. 물론,

이를 위해 평가자의 역할이 필수적이다.

평가자의 역할이 필수적이긴 해도, 그 의사 결정 자체가 플랜트에 학습 정보를 직접 제공하지는 않는다. 다만, 평가자는 CMAC에 의한 제어 결과를 평가하고 그에 따른 CMAC을 위한 적절한 학습 정보 제공처를 선정해 줄 뿐이다. 또한, 의사 결정 순서도 중요한 인자이다. 두 퍼지 제어기의 활용 방법에 따른 시스템 행태를 관찰한 바, 막대의 기울기가 더 커졌는지를 우선적으로 관찰하는 것이 긴요하다. 이것은 2절에서 살핀 운반차-막대 시스템의 제어와 관련된 일반적인 특성과 상통하는 사실이다. 막대가 더 기울어 졌다면 CMAC이  $\phi$ -학습을 하도록 스위치를 설정하여야 한다. 막대가 더 기울어지지 않은 것을 확인한 다음, 운반차의 위치를 따지는데, 만약 운반차가 트랙 중앙으로부터 더 멀어졌다면, 그 상태에 대해 CMAC이  $x$ -학습을 하도록 스위치를 설정한다. 매 제어 단계에 대한 학습률은, 1로 정하였다. 이 우선순서를 달리하면 CMAC은 전혀 다른 훈련 과정을 거치게 되어, 제어계의 특성이 전혀 달라질 것이다.

그러므로, CMAC은 자신의 출력에 의해 플랜트가 적절히 운전되지 않았을 때, 학습의 기회를 가지게 된다. CMAC의 학습 간섭[15] 문제를 생각할 때 이 특성은 바람직한 것이다. 만약 올바른 출력을 산출했음에도 불구하고 학습을 거듭한다면, 바로 학습에 혼란을 초래하게 되는 것이다. 그러나, 실패한 경우에 대해서 학습 기회를 제공함으로써 학습 간섭의 문제를 줄일 수 있는 훈련 방법이라고 할 수 있다.

또한, 연이어지는 제어 시각에 대한 플랜트의 두 상태는 중간변수 공간에서 서로 이웃한 상태들일 가능성이 크다. 그러므로, CMAC이 어떤 상태에 대해 학습한 효과가 다음 제어 시각에 대한 상태에 대해 곧 바로 나타날 것이다.

그림 8은 그림 6의 제어계에 의한 시스템 행태를 보여준다. CMAC은 전혀 훈련을 거치지 않은 상태였고, 초기 조건은, 그림 3과 그림 5에서와 같이, (1.0, 0.4, -9.5, -56.0) 였고, 트랙의 범위는 [-2.4, 2.4]였다. 세 그림을 비교하면, 그림 8의 시스템 행태가  $x$ -제어기나  $\phi$ -제어기에 의한 시스템 행태와는 전혀 다른 것을 알 수 있다. 막대의 각도가 불연속적으로 변동하는 현상이 보이지만,  $x$ -제어기에 의한 것 보다, 시스템이 훨씬 빨리 안정되어 제어 목적을 달성하는 것을 알 수 있다.

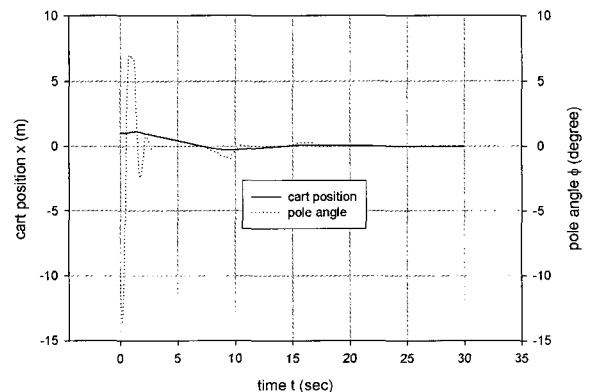


그림 8. 그림 6 제어 계략에 의한 시스템의 행태.  
Fig. 8. System behavior due to the control scheme of Fig. 6.

### 5. 토 의

그림 8에서 보는 시스템 행태는 그림 3이나 그림 5에서 보는 것과는 전혀 다르다. 그러나 그림 8의 시스템 행태를 나타내는 그림 6의 제어기는 그림 3과 그림 5의 시스템 행태를 나타내는 두 퍼지 제어기에 바탕을 두고 있다는 사실에 주목하여야 한다. 즉, 일부 상태 변수에 의해 정의된  $x$ -제어기나  $\phi$ -제어기 각각은 운반차-막대 시스템을 적절히 제어할 수 없음에도 불구하고, 그런 두 제어기를 모델로 학습한 CMAC에 의해 결과적으로 새로운 제어 역량을 생성하는 제어기가 개발된 것이다.

표 3. 연이은 제어 시각에 대해 CMAC이 학습한 제어기와 시스템에 작용한 힘.

Table 3. Switch selection for CMAC and forces applied to the plant for some control steps

t	x	$\dot{x}$	$\phi$	$\dot{\phi}$	switch	force
9.46	-0.30168	-0.0182	-0.9488	1.2559	x	1.2691
9.48	-0.30185	0.0006	-0.9371	-0.0828	x	1.2490
9.50	-0.30193	-0.0084	-0.9282	0.9711	c	1.2676
9.52	-0.30185	0.0161	-0.9271	-0.8621	p	-0.6352
9.54	-0.30137	0.0324	-0.9556	-1.9870	p	-0.9595
9.56	-0.30085	0.0193	-0.9812	-0.5746	c	-0.8947
9.58	-0.30065	0.0015	-0.9746	1.2357	c	0.0000

그림 8에 대해서, 주어진 초기 상태에 대해서, CMAC은 아무 사전 훈련도 없는 상태로 50 초 동안  $\Delta t = 0.02$ 에 대해,  $\phi$ -학습 1,317 회와  $x$ -학습을 213 회 한 반면, 971 회 동안은 적절한 출력을 생산하였다. 몇 시각 단계에 대해, 학습 상황을 살펴보자. 표 3에서, 'switch'의 x는 CMAC이  $x$ -제어기의 출력을, p는  $\phi$ -제어기의 출력을 학습하였다는 뜻이고, c는 그 시각에 CMAC은 학습 없이 자신의 출력으로 시스템을 제어한 것을 의미한다. 예를 들어,  $t = 9.46$ 에서 CMAC이 플랜트로 보낸 출력에 의해  $t = 9.48$ 에서 막대의 각도  $\phi$ (도)는 -0.9488에서 -0.9371로 커지지 않았지만, 운반차가 트랙 중앙으로부터 더 멀어져서  $x$ 가 -0.30168에서 -0.30185가 되었다. 그래서 스위치가  $x$ -학습으로 설정되어, CMAC은 'force' 1.2691을 학습하였다. 다음 시각 단계  $t = 9.48$ 에서도 CMAC의 출력은 적절하지 못하여 다시  $x$ -학습을 하였다. 그러나,  $t = 9.50$ 에서는 앞 시각 단계에서  $x$ -학습의 결과로 CMAC이 적절한 힘 1.2676을 제공하여,  $\phi$ 는 -0.9282에서 -0.9271로,  $x$ 도 -0.30193에서 -0.30185로 줄었다. 그런데,  $t = 9.48$ 에서의 상태와  $t = 9.50$ 에서의 상태는 CMAC에서 변수  $x$ 와  $\phi$ 에 대해서 같은 중간변수를 가지므로 서로 이웃 상태로서, 앞 시각 단계에 대한 학습 효과가 다음 시각 단계에 대한 제어에서 곧 바로 나타난 것을 확인할 수 있다.

그림 9는 운반차-막대 시스템을 위한 비선형 제어 법칙 [17]에 의한 시스템 행태를 보여준다. 그림 8을 그림 9와 비교하면, 이 논문에서 개발한 제어기는 막대의 각도 변화가 느리며 그에 따른 요동이 상대적으로 조금 더 오래 지속되어 시스템이 안정되기 까지 시간이 더 걸리는 점에서, 해석적 방법 보다는 좋지 않은 제어 특성을 보인다. 그러나, 이 논문

에서 제시하는 CMAC 훈련에 의해, 결과적으로 두 퍼지 제어기가 협동해서 플랜트를 제어하는 계략은 흥미롭다. CMAC이 출력을 낸 결과가 실패로 평가되면 그에 대한 불완전한 학습 정보를 퍼지 제어기로부터 구하지만, CMAC의 훈련과 제어 계략의 특성에 의해 CMAC은 적절한 학습 효과를 나타내는 것이다.

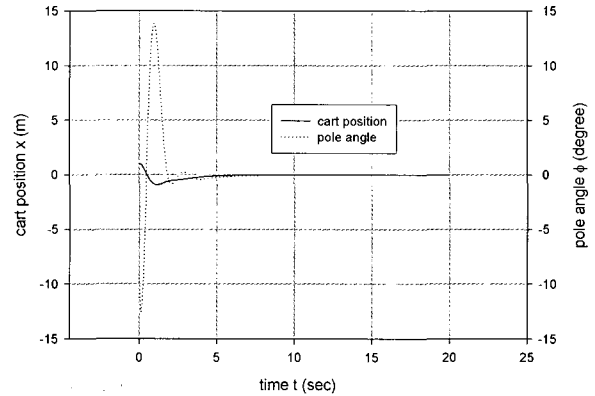


그림 9. 비선형 제어 법칙[17]에 의한 시스템 행태.  
Fig. 9. Cart position and pole angle versus time for a nonlinear control law [17].

뿐만 아니라, 4 개의 상태 변수 값을 제어에 반영하기 위해 다른 매개변수를 설정하거나[6] 시스템 상태를 구분하는 [7] 과외의 계산 부담 없이 상태 변수를 그대로 입력 변수로 사용하므로 제어기를 쉽게 설계할 수 있다. 또한, CMAC 순람표에서 제어를 위한 출력 값을 바로 가져다 쓰는 것임으로, 평가자의 판단에 소요 되는 시간 외에, 과외의 계산 부담은 없다. 이런 장점들이 이 논문에서 제시하는 제어 계략을 평가하는 중요 요소이다.

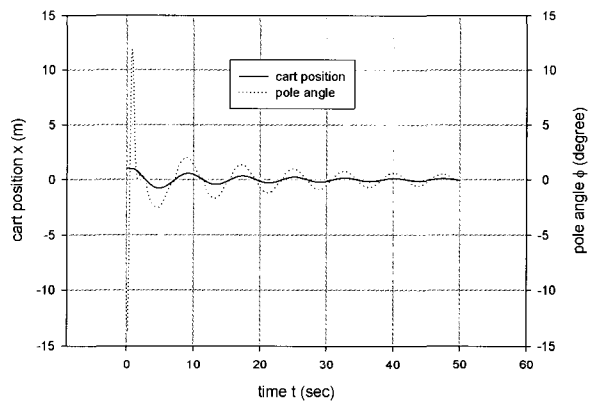


그림 10. 4 차원 퍼지 제어기에 의한 시스템 행태.  
Fig. 10. Cart position and pole angle versus time for the fuzzy controller with 4 input variables.

그림 10은 그림 2의 소속 함수와 표 1과 표 2의 제어 규칙과 유사한 방법으로 구성된 입력 변수 4 개로 구성된 퍼지 제어기에 의한 운반차-막대 시스템의 행태를 보여준다. 시스템이 안정되는데 시간이 꽤 오래 걸리는 것을 알 수 있다. 이 그림을 그림 8과 그림 9에 비교해 보면, 4 차원 퍼지 제어기에 의한 시스템 행태보다는, 이 논문에서 제시하는 제어 계략에 의한 시스템 행태(그림 8)가 해석적 방법에 의한 시스템

템 행태(그림 9)와 더 유사함을 알 수 있다. 제어기 구성 관점에서, 4 차원 퍼지 제어기의 경우에, 무려 81 개나 되는 제어 규칙을 사용해야 하지만, 2 차원에 대해서는 각 퍼지 제어기에 9 개의 제어 규칙이 필요할 뿐이다.

CMAC 구축에 영향을 미치는 매개변수  $K$ 의 작은 값과 큰 값에 대해서 여러 다양한 제어계를 시험해 보았으나,  $K$ 의 변화가 시스템 행태에 미치는 현저한 영향은 발견하지 못하였다. 다만, 연속적인 입력 변수  $x$ 와  $\phi$  축의 원점 주위로 조밀하게 양자화하는 것은 시스템이 더 빨리 안정되게 하는데 영향을 나타내었다.

제어 목표에 도달하기 까지 상대적으로 긴 시간이 요구되는 상태, 예를 들어,  $(1.0, 10.0, 8.0, -20.0)$ 에 대해서, 그림 10의 제어기의 경우 운반차가 트랙을 따라 16 m까지 멀어졌다가 50 초 내에 시스템이 안정되었지만, 이 논문에서 개발한 제어계는 운반차를 트랙을 따라 33 m까지 끌고 갔다가 140 초가 걸려 시스템을 안정하였다. 그렇지만, 이 논문에서 개발한 제어계는 4 차원 퍼지 제어기보다는 비선형 제어 법칙에 의한 것과 유사한 시스템 행태를 나타낸다.

다양한 상태에 대해서 제어 경험을 쌓은 뒤의 제어계에 의한 시스템 행태를 살피기 위해서 다음과 같은 5 회의 제어 경험을 가진 뒤, 초기 상태에 대한 제어계의 시스템 제어 행태를 살펴본다: ①  $(1.0, 0.4, -9.5, -56.0)$ , ②  $(-2.0, 1.0, 0, 0)$ , ③  $(2.0, -1.0, 0, 0)$ , ④  $(1.0, 1.0, 10.0, -10.0)$ , ⑤  $(1.0, 1.0, -10.0, 10.0)$ . 그랬더니, 시스템 행태는 그림 8과 흡사했고, CMAC은  $\phi$ -학습을 1,211 회,  $x$ -학습을 246 회 하였으며, 1,044 회 동안은 적절한 출력을 하였다. 이것을, 다른 상태에 대한 제어 경험 없이는 CMAC이 971회의 적절한 출력을 내보냈던 것과 비교하면, CMAC의 제어 능력이 약간 나아진 것으로 평가할 수 있다. 다른 상태들에 대한 제어 경험의 누적에 항상 CMAC의 제어 능력을 향상시키는 것은 아니었지만, 대체적으로 그런 성향을 나타내는 것을 관찰할 수 있다.

여기서, CMAC이 퍼지 제어기가 생산하는 운반차-막대 시스템에 가하는 힘을 학습하는 것에 대해 생각해 보자. 퍼지 제어기는 연속적인 출력 값을 생산하는데 비해서 CMAC의 출력 값은 그렇지 못하다. 그러므로, 지난 상태에 대해 적절한 출력 값을 학습하기는 하지만 대부분의 경우에 제어 시각의 출력은 CMAC 학습 결과에 의한 것이다. 그래서, 시스템이 안정되기 까지 더 오랜 시간이 걸린다고 볼 수 있다. 그리고 이것은 CMAC의 학습 특성에 기인하는 것이 아니고 제어 계략에 기인하는 것이다.

## 6. 결 론

두 기의 2 차원 퍼지 제어기가 외부 감독에 의한 CMAC의 학습 계략에 의해 협동하게 하는 운반차-막대 시스템을 제어하는 계략을 개발하였다. 운반차의 변위와 그 속도를 2 개의 입력 변수로 하는 퍼지 제어기와 막대의 각도와 각속도를 2 개의 입력 변수로 하는 퍼지 제어기 각각은 운반차의 위치와 막대 각도를 적절하게 제어하지 못하지만, 두 제어기가 CMAC의 학습 결과에 의해 협동하는 제어계는 운반차-막대 시스템을 성공적으로 제어하였다.

이 논문에서 제시한 제어 계략이 해석적 방법에 의한 제어에 비해 성능이 낮지는 않지만, 4 개의 입력 변수에 의한 4 차원 퍼지 제어기에 비해 더 나은 성능을 보였다. 또한, 과거의 계산을 수반하는 다른 매개변수들을 이용하는 기법들에

비해서는, 다른 계산 없이 CMAC의 순람표에 의한 출력 변수의 즉각적인 산출이 계산 시간 면에서 장점으로 평가될 수 있다.

더욱이 제어계가 운반차-막대 시스템에 가할 힘을 학습하기 보다는, 시스템의 상태에 적절한 퍼지 제어기를 선정하는 스위치 기능을 학습하는 것이 더 나은 계략이 될 것 같다. 즉, 4 개의 상태 변수에 의해 정의되는 스위치를 구현할 지능 요소는 경험을 통해 시스템의 상태에 적합한 퍼지 제어기를 고르는 일을 학습하게 될 것이다. 그러면, 그 지능 요소의 출력에 따라 둘 중 어느 하나의 퍼지 제어기 출력이 운반차-막대 시스템에 작용하게 될 것이다.

이 연구에서 개발한 두 퍼지 제어기의 협동 제어 계략은, 시스템의 직선 운동 특성에 관련된 변수와 회전 운동 특성에 관련된 변수들이 시스템 제어 행태에서 구별되게 나타나는 특별한 문제에 대한 것이기는 하지만, 그와 유사한 다른 문제들에 대해서도 같은 기법을 적용할 수 있을 것이다. 예를 들어, ball-beam 시스템의 제어에도 적용될 수 있을 것으로 사료된다.

## 참 고 문 헌

- [1] S. Geva and J. Sitte, "A Cartpole Experiment Benchmark Trainable Controllers," *IEEE Control Systems*, pp. 40-51, October 1993.
- [2] K. M. Passino, *Biomimicry for Optimization, Control, and Automation*, Springer-Verlag London Limited, pp. 222-223, 2005.
- [3] C-C Wong, S-M Feng, "Switching-Type Fuzzy Controller Design by Genetic Algorithms," *Fuzzy Sets and Systems*, Vol. 74, pp. 175-185, 1995.
- [4] C. L. Karr, *Practical Applications of Computational Intelligence for Adaptive Control*, CRC Press, 1999.
- [5] S. Kawaji and T. Maeda, "Fuzzy Servo Control System for an Inverted Pendulum," *Proceedings of IFES '91*, Vol. 2, pp. 812-823, 1991.
- [6] J. Yi and N. Yubazaki, "Stabilization Fuzzy Control of Inverted Pendulum Systems," *Artificial Intelligent in Engineering*, Vol 14, pp. 153-163, 2000.
- [7] T. S. Li and M. Shieh, "Switching-Type Fuzzy Sliding Mode Control of a Cart-Pole System," *Mechatronics*, Vol. 10, pp. 91-109, 2000.
- [8] C. E. Lin and Y. Sheu, "A Hybrid-Control Approach for Pendulum-Car Control," *IEEE Transactions on Industrial Electronics*, Vol. 39, No. 3, pp. 208-214, June 1992.
- [9] K. Tanaka, *An Introduction to Fuzzy Logic for Practical Applications*, Springer, 1997.
- [10] H. Lai and C. Wong, "A Fuzzy CMAC Structure and Learning Method for Function Approximation," *2001 IEEE International Fuzzy Systems Conference*, pp. 436-439, 2001.
- [11] X. Xu, D. Hu and H. He, "Accelerated Reinforcement Learning Control using Modified CMAC Neural Networks," *Proceedings of the 9th International Conference on Neural Information*

*Processing (ICONIP'02)*, Vol. 5, pp. 2575-2578, 2002.

- [12] H. Wei, W. Yong, M. Xuanqin and W. Yan, "A Cooperative Fuzzy Control Method for Traffic Lights," *Proceedings of the 2001 IEEE Intelligent Transportation Systems Conference*, August 25-29, pp. 185-188, 2001.
- [13] J. Liu, D. Liu, G-H Zhang, and P-S Wu, "An Optimal Design of Fuzzy Controller for High Power Oil Transfer System," *Proceedings of the First International Conference on Machine Learning and Cybernetics*, November 4-5, pp. 699-704, 2002.
- [14] A. B. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 5, pp. 834-846, September/October 1983.
- [15] 권성규, "CMAC을 위한 이웃간훈련 방법," *대한기계학회논문집*, 제 16 권 10 호, pp. 1816-1823, 1992.
- [16] 권성규, "운반차-막대 시스템의 대칭성과 Table Look-Up 제어 기법," *한국 퍼지 및 지능 시스템 학회논문지*, 제 14 권 3 호, pp. 290-297, 2004.
- [17] A. Guez and J. Selinsky, "A Trainable Neuromorphic Controller," *Journal of Robotic Systems*, Vol. 5, No. 4, pp. 363-388, 1988.

## 저 자 소 개



권성규(Sunggyu Kwon)

1980년 : 연세대학교 기계공학 학사

1983년 : 연세대학교 기계공학 석사

1990년 : Louisiana State University, 기계공학 박사

1991년 : 한국원자력연구소 선임연구원

1995년 : 계명대학교 기계자동차공학부 부교수

관심분야 : CMAC, 지능제어, 기구 설계

Phone : (053) 580-5231

Fax : (053) 580-5116

E-mail : cmack@kmu.ac.kr