

# 디지털 시뮬레이션에 의한 CMAC 신경망 직류전동기 속도 제어기 설계

崔 光 皓, 趙 鏞 範

## Design for CMAC Neural Network Speed Controller of DC Motor by Digital Simulations

Kwang-Ho Choi, Yong Beom Cho

### 요 약

본 논문에서는 비선형 시스템을 제어하기 위한 CMAC 신경망을 제안한다. CMAC 신경망은 사람의 소뇌를 모방한 신경망으로서 복잡한 비선형 함수의 해를 수치적인 연산에 의해 구하지 않고 table Look-up 방식을 이용하기 때문에 학습이 타 신경망에 비해 월등히 빠르고 용이하며 제어신호를 출력하기 위한 계산시간이 거의 필요치가 않다. 본 논문에서는 제안한 제어기 구조의 타당성을 증명하기 위해 간단한 비선형 함수와 직류전동기 속도제어에 대한 CMAC 제어기를 시뮬레이션을 통하여 학습 제어기의 안정성 및 추적에러의 감소를 확인하였다. 또한 제안 CMAC 제어기를 실시간 장력제어에 적용하여 직류전동기의 속도를 제어하므로 시뮬레이션 값과 비슷한 장력제어를 보임으로서 유용성을 입증하였다.

### ABSTRACT

In this paper, we propose a CMAC(Cerebellar Model Articulation Controller) neural network for controlling a non-linear system. CMAC is a neural network that models the human cerebellum. CMAC uses a table look-up method to resolve the complex non-linear system instead of a numerical calculation method. It is very fast learn compared with other neural networks. It does not need a calculation time to generate control signals. The simulation results show that the proposed CMAC controllers for a simple non-linear function and a DC Motor speed control reduce tracking errors and improve the stability of its learning controllers. The validity of the proposed CMAC controller is also proved by the real-time tension control.

**Key Words** : CMAC Neural Network, DC Motor, Speed Control, Tension Control, Non-linear control

### 1. 서 론

선형적인 시스템의 경우 많은 시스템 제어 이론들이 제안되었으며, 성공적으로 사용되고 있다. 하지만 실제 물리적인 시스템은 비선형성을 내포하고 있으며, 이러한 비선형 시스템을 제어하기 위해 선형 수학 모델을 이용할 경우 모델에 불확실성이 존재하게 되어 오차가 발생하여, 정확한 시스템의 제어와 식별이 불가능하다. 따라서 비선형 시스템을 제어하기 위해 오차가 허용될

수 있는 범위를 한정하고 동작점 부근에서의 선형 근사화를 통하여 제한적으로 시스템을 규명하여 이를 제어하는 방법을 사용한다. 이러한 방법은 지금도 대부분의 산업 시스템에 사용되고 있으며, 매우 성공적인 결과를 보여주고 있다. 하지만 이러한 접근방법은 시스템의 파라미터 변화와 같은 동적 특성의 변화와 작업환경의 변화에 대한 적응능력이 떨어지고 작업조건이나 제어환경이 바뀔 때마다 제어기를 다시 설계하거나 파라미터를 재조정해야 하는 단점이 있다. 이러한

고전적인 제어방식이 갖는 문제점을 해결하기 위한 많은 비선형 시스템 제어 이론들이 제안되고 있다. 그 중에 가장 주목을 받는 제어 이론 중의 하나가 신경망을 이용한 시스템의 제어 방법이다.

신경망은 학습을 통하여 선형 혹은 비선형 함수를 근사화 할 수 있다. 이는 기존의 제어 기법에 비해서 시스템의 파라미터를 추출하는데 필요한 복잡한 수식이나 모델이 필요하지 않으며, 강한 비선형 시스템도 성공적으로 제어할 수 있기 때문에 신경망을 비선형 시스템의 규명과 제어에 적용하려는 연구가 폭넓게 진행중이다. 시스템의 제어라는 관점에서 가장 많이 사용되고 있는 신경망 구조는 다층 신경망 구조이다. 다층 신경망은 학습을 통하여 자신의 가중치 값을 변화시켜 시스템의 입출력 관계를 모사하고 이를 통해 시스템의 비선형 특성을 모사할 수 있다. 3층 이상의 다층 신경망 구조는 오차 역전파 알고리즘을 이용할 경우 비선형 시스템을 어떤 원하는 정확도까지 근사화할 수 있다<sup>[1]</sup>. 그러나 로봇 제어와 같이 입력이 다차원인 문제나 강한 비선형 시스템의 제어 문제 등에 다층 신경망을 이용하는 경우, 입력 패턴의 변화에 대한 학습이 어렵고 빠른 속도의 동작이 힘들며, 중간층의 수와 뉴런의 수 등의 구조에 대한 정확한 해가 없기 때문에 실용적 응용에서의 성공 사례가 많지 않다.

본 논문에서는 비선형 시스템을 식별하고 이를 제어하기 위해 CMAC(Cerebellar Model Articulation Controller) 신경망을 사용한다. CMAC 신경망은 사람의 소뇌를 모방한 신경망으로서 복잡한 비선형 함수의 해를 수치적인 연산에 의해 구하지 않고 Table Look-up 방식을 이용하기 때문에 학습이 타 신경망에 비해 월등히 빠르고 용이하다. 또한 제어신호를 출력하기 위한 계산시간이 거의 필요치가 않기 때문에 로봇 제어 등에 적용한 사례들이 많다<sup>[2-5]</sup>.

직류전동기 제어에 인공지능 기법을 이용한 예는 있으나 대부분 퍼지룰<sup>[6]</sup>이나 역전파 알고리즘을 사용하였다. 본 논문에서는 CMAC 신경망 제어기의 타당성을 증명하기 위해 간단한 비선형 함수와 직류전동기에 대한 모의 실험을 하였다. 또한 웹 이동시스템에서의 직류전동기 속도 제어로 인한 장력제어를 실제적인 예로 소개한다.

## 2. CMAC 신경망

1970년대 Albus에 의해서 개발된 CMAC 신경망은 빠른 학습 속도와 하드웨어 구현의 용이성 등의 특성 때문에 학자들의 관심이 점차 높아지고 있는 신경망이

다<sup>[7]</sup>. 지금까지 많은 학자들이 CMAC에 대하여 연구를 하였으며 그 결과, Wong과 Sideris는 CMAC 학습이 훈련 데이터 집합에 대하여 임의의 정확도 이내로 항상 수렴함을 증명하였고, Cotter와 Guillerm은 CMAC이 구조적으로 Kolmogorov 이론으로부터 유도된 신경망과 유사함을 보이는 등 많은 이론적 발전이 있었다. 또한 CMAC 신경망의 경우 이론적인 내용보다 오히려 응용분야에 있어서의 연구가 활발히 진행중이다. 그 중에서 가장 두드러지게 기여한 사람은 Miller<sup>[2-5]</sup>였다. 그는 동적균형(Dynamic Balance)을 갖는 2족 보행 로봇에 CMAC을 적용하여 인상적인 연구결과를 발표하였다. 이후에도 다양한 로봇 제어에 CMAC을 적용시킴으로써, 신경망을 이용한 제어 분야에 있어서 CMAC의 위치를 공고히 하였으며 패턴 인식, 영상 처리 등 신호처리 분야에도 CMAC을 적용할 수 있다는 가능성을 보여 주었다. CMAC은 인간의 소뇌 구조를 모방한 신경망으로 매우 간단한 구조를 가지고 있다. 그림 1은 Albus가 제안한 개념적인 CMAC의 구조를 나타내고 있다. 그림 1에서 보는 바와 같이 CMAC은 다음과 같은 3가지 단계로 이루어져 있다.

- 첫째, 주 제어기 혹은 센서로부터의 정보는 양자화되고 개념적 메모리(Conceptual Memory)에 대한 주소로 사용된다.
- 둘째, CMAC에서 상태에 대한 정보는 메모리에 분산되어 저장되고 주소의 키(Key)로써 상태변수를 이용하여 검색하게 된다.
- 셋째, 데이터 검색을 위해 상태변수들은 중간변수들로 매핑된 후 출력값을 검색하기 위해 물리적 메모리 주소로 매핑된다.

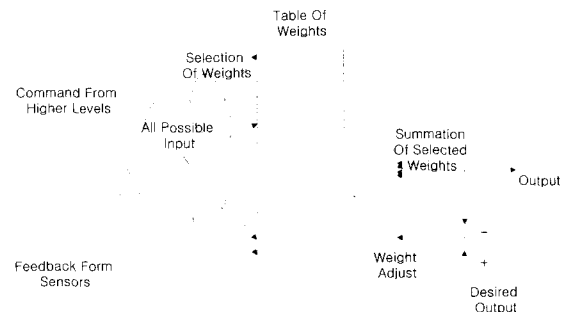


그림 1 Albus의 CMAC 모델  
Fig. 1 CMAC model of Albus

CMAC의 자세한 구조와 그 학습 방법은 그림 2에서와 같이 CMAC의 입력 S는 먼저 임의의 개수로 양

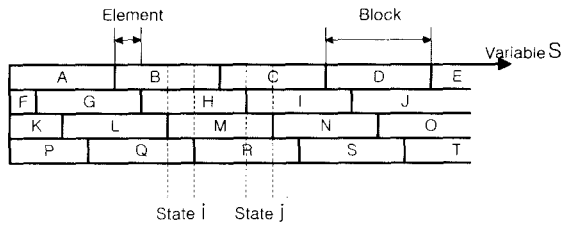


그림 2 단일 입력 변수에 대한 CMAC 블록의 변환  
Fig. 2 Conversion of CMAC block for one-input variable

자화 된다. 그리고 양자화된 입력 S는 블록(Block)이라는 여러 개의 영역으로 분할되고 각 블록은 소자(Element)라고 불리는 부영역으로 분할된다. 또한 입력변수 S는 블록의 크기만큼 만들어져야 하는데, 이는 그림 2에서 세로방향의 깊이를 의미한다. 이 때 중요한 점은 각 제너레이션(Generation) 또는 각 깊이마다 블록들은 1 소자 크기만큼 옆으로 이동되어 시작하기 때문에 제너레이션의 위치에 따라 블록의 경계위치가 달라지게 된다. 각 제너레이션에서의 블록들은 서로 다른 인식명(Identification Name)을 할당받는다. 그러므로 양자화된 하나의 상태는 여러 개의 층, 즉 제너레이션에 속하는 블록명의 집합으로 특징화 될 수 있다. 예를 들어 그림 2에서 상태 i와 j는 각각 (B,H,M,Q)와 (C,I,M,R)에 의해 특징화한다. 여기에서 블록은 실제적인 메모리를 하나씩 할당받게 되고, 학습 데이터를 저장한다.

어떤 입력이 들어올 경우 입력은 여러 개의 블록을 지정하게 되고 각각의 블록에 정보를 분산하여 저장하게 된다. 위의 CMAC 정의를 수식적으로 표현해 보면 다음과 같다. 먼저 N차원의 실제 입력이 들어온다고 가정하면 입력벡터 S는 식 (1)과 같다.

$$S = \langle s_1, s_2, \dots, s_N \rangle \quad (1)$$

여기서 제너레이션 인자(하나의 입력에 대해 동시에 여기 되는 메모리 수)는 C로 가정한다. 먼저 입력 벡터를 정규화하기 위해서 적당한 양자화 파라미터  $\Delta_i$ 로 나눈다.

$$S = \langle s_1, s_2, \dots, s_N \rangle$$

$$= \langle \text{int}\left(\frac{s_1}{\Delta_1}\right), \text{int}\left(\frac{s_2}{\Delta_1}\right), \dots, \text{int}\left(\frac{s_N}{\Delta_1}\right) \rangle \quad (2)$$

위의 수식에서 알 수 있듯이 각 블록의 폭은  $C \times \Delta_i$ 와 같으며, 이를 감응 영역(Receptive Field)이라고 정의한다. 다음으로 입력벡터 S에 의해 지정된 감응 영역의 벡터 주소  $A_i$ 를 구한다.

$$A_i = \langle s_1 - ((s_1 - i) \% C), s_2 - ((s_2 - i) \% C)$$

$$\dots, s_N - ((s_N - i) \% C) \rangle$$

$$= \langle a_{i1}, a_{i2}, \dots, a_{iN} \rangle \quad i=1, 2, \dots, C \quad (3)$$

여기서, %는 나머지(modulus) 연산자를 나타내고 i는 전체가 C개의 층인 감응 영역 층의 인덱스를 나타낸다. 그러나 위의 방법으로 메모리를 지정할 경우 제너레이션이 많거나 입력벡터의 수가 증가할 때, 필요한 메모리의 개수도 크게 증가하게 된다. 따라서 메모리를 줄이기 위한 노력이 필요하며 대부분의 CMAC 알고리즘에서는 의사 랜덤함수를 이용한 하쉬코딩(Hash Coding)을 이용하여 메모리의 사용량을 줄이고 있다. 식 (3)을 하쉬코딩을 이용하여 다시 정의하면

$$A_i = h(a_{i1}, a_{i2}, \dots, a_{iN}) \quad (4)$$

결과적으로 CMAC의 출력  $y(S)$ 는 입력벡터에 의해 지정된 감응 영역에 대응하는 메모리들의 값을 평균하여 출력한다.

$$y(S) = \frac{1}{C} \sum_{i=1}^C W[A_i] \quad (5)$$

그리고 상태 S에 학습시켜야 할 목표값을  $y_d(S)$ 라고 하면  $y_d(S)$ 와  $y(S)$ 의 차이값에 비례하는 갱신값  $\Delta W$ 는 다음과 같이 계산된다.

$$\Delta W = \beta ( y_d(S) - y(S) ) \quad (6)$$

여기서  $\beta$ 는 학습계수를 나타낸다. 이 갱신값은 현재 출력을 만들기 위해 사용된 메모리에 다시 가감되어 메모리를 갱신하게 된다. 따라서 다음에 같은 입력이 들어올 경우 출력은  $y_d(S)$ 에 가깝게 출력하게 된다. 정리하면 CMAC에서 상태에 대한 정보는 메모리에 분산되어 저장되고 주소의 key로써 상태변수를 이용하여 검색하게 된다.

데이터 검색을 위해 상태변수들은 중간변수들로 사상(mapping)된 다음, 출력값을 검색하기 위해 물리적 메모리 주소로 사상된다.

### 3. CMAC 제어기 실험

모의 실험에 사용할 비선형 시스템의 모델은 다음과 같다.

$$Y(k+1) = 0.5Y(k) + \sin[Y(k)] + U(k) \quad (7)$$

여기에서  $Y(k)$ 는 시스템의 출력,  $U(k)$ 는 제어 입력을 의미한다. 이 비선형 모델은 Chen<sup>[8]</sup>이 CMAC 신경망의 안정성을 검증하기 위해 사용했던 모델이며 되먹임 오차 학습법의 대표적인 형태이다. Chen은 되먹임 오차 학습법을 이용한 CMAC 제어기 모델이 초기에는 안정하지만 갑자기 불안정한 상태에 빠지게 됨을 지적하고 이를 개선하기 위해 Dead-Zone을 갖는 학습 규칙을 제안하였다. 본 논문에서 시뮬레이션을 위해 사용한 제어기 구조는 그림 3과 같으며 Chen의 제어기 모델과 거의 동일하다.

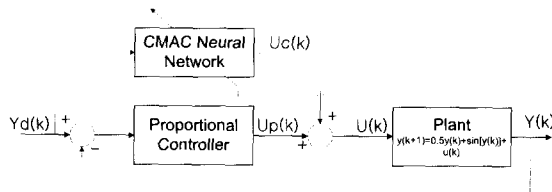


그림 3 CMAC 제어기  
Fig. 3 CMAC control system

제어기에서 CMAC은 P 제어기(Proportional controller)의 출력과 자신의 출력의 합을 학습하게 되며  $Y_d(k)$ 와  $Y(k)$ 가 최소가 되도록 메모리를 갱신한다. 제어하려는 플랜트(Plant)의 비선형 관계식이 매우 간단하기 때문에 PID제어기 대신에 P 제어기를 사용하였다. CMAC 신경망의 파라미터와 P 제어기의 이득은 표 1과 같이 주어졌다.

표 1 모의 실험을 위한 제어 파라미터  
Table 1 Control parameters for simulation

Reference Signal	$\sin(2\pi k / 100)$
Sampling Rate	1
Proportional Gain	1.4
Learning Rate	0.01
Quantization	199
Generation	9

그림 4는 P 제어기만을 사용했을 경우의 입력과 출력을 나타낸다. 제어 대상이 비교적 단순한 비선형 시스템이기 때문에 출력신호가 입력 기준 신호를 빠르게 추종하고 있다. 하지만 약 0.2 정도의 지속적인 Tracking Error가 발생하고 있어 정확한 제어를 보장할 수 없다. 그림 4에서의 가로축은 [sec]를 의미하며 1/60 [sec]가 한 주기의 정현파를 만들고 100번의 Time Steps가 경과함을 나타낸다.

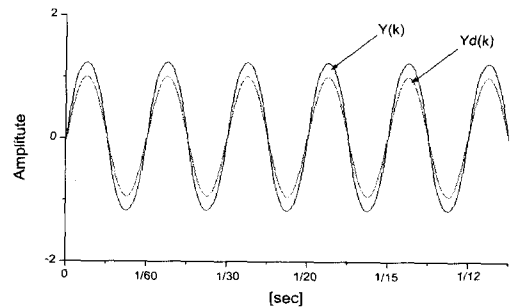


그림 4 P 제어기를 사용한 입력과 출력파형  
Fig. 4 Input and output waves using P controller

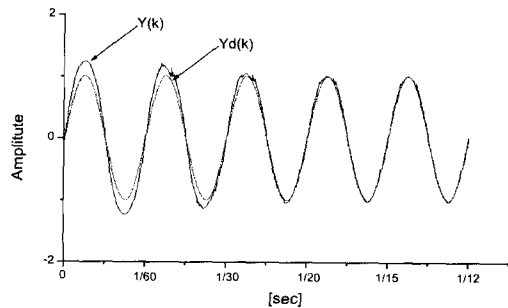


그림 5 CMAC 신경망 제어기의 입력과 출력파형  
Fig. 5 Input and output waves of CMAC neural network controller

그림 5에서는 P 제어기와 CMAC을 같이 사용한 경우이며, 정현파의 두 번째 주기부터 CMAC 신경망이 동작하도록 하였다. CMAC 신경망이 추가되었을 때, Tracking Error는 0.013까지 급격하게 줄어드는 것을 볼 수 있다.

그림 6은 P 제어기의 제어신호  $U_p$ 와 CMAC 신경망의 제어 신호  $U_c$ 의 변화를 보여주고 있다. 그림 6에서 보여주는 바와 같이 초기 상태에서는  $U_p$ 가 지배적인 신호로 작용하고 있으나 CMAC 신경망이  $U_p+U_c$  신호를 학습함에 따라서  $U_c$ 신호가  $U_p$ 신호를 추종하여 점점 증가되고 있다. 가로축은 [sec]를 나타내며, 1/60의 인자를 곱하여야 한다.

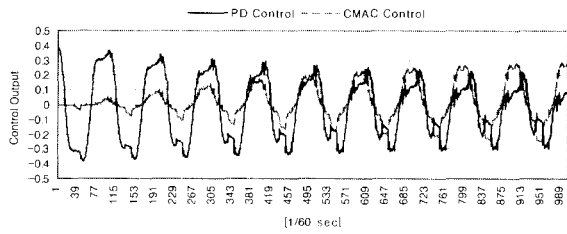


그림 6 매니플레이터의 제어 신호  
Fig. 6 Control signal of manipulator

#### 4. 직류전동기 속도제어를 위한 CMAC 제어기 설계

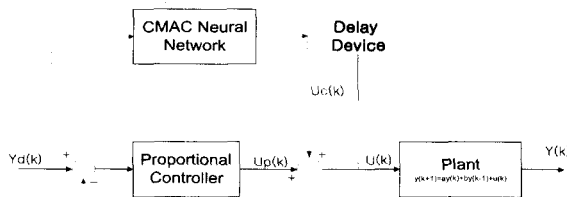
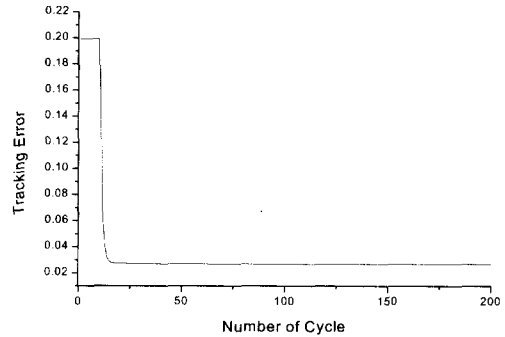


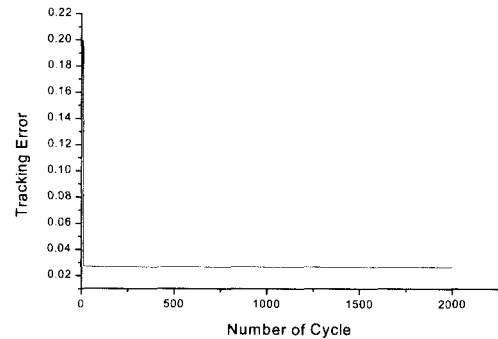
그림 7 직류전동기 속도제어를 위한 CMAC 신경망 제어기  
Fig. 7 CMAC Neural network controller for speed control of DC motor

직류전동기 속도제어를 위한 CMAC 제어기는 Sampling Rate를 빠르게 하여 수렴율을 개선할 필요가 있다<sup>[9]</sup>. 하지만 실제적으로 신경망 처리기의 시간 지연, 센서의 응답 지연 시간 등으로 인해 Sampling Rate의 지속적인 증가는 어렵기 때문에 그림 7과 같이 CMAC 신경망의 신호를 지연시키는 제어기 구조를 제안한다. 그림 7의 구조는 일반적인 간접 오차 역전달 방식의 신경망 제어기와 크게 다르지 않다. 다만 CMAC 제어기의 출력에 지연소자를 첨가함으로써 현재의 출력이 바로 다음의 CMAC 제어기 출력에 영향을 주지 않고 몇 단계 후의 출력에 영향을 주게 하였다. 이는 CMAC 제어기 출력의 갑작스런 변화를 막고 P 제어기와의 발진 현상을 완화시켜 준다. 그림 8은 CMAC 신경망의 출력을 4개까지 적분하여 그 평균 출력을 제어신호로 사용한 경우이다. 앞에서의 모의 실험에서와 마찬가지로 10번째 Sin주기에서 CMAC 신경망 제어기를 추가하였으며 다른 파라미터는 표 1을 따르고 있다. 지연소자를 제외한 CMAC 신경망 제어기에서는 100번에서 166번의 주기에서 발산 현상이 나

타하며, 지연소자를 첨가하므로써 2000번 이상의 주기에서도 발산현상이 나타나지 않음을 볼 수 있다.



(a) 제안한 CMAC 제어기의 평가 (I)  
(a) Evaluation (I) of proposed CMAC controller



(b) 제안한 CMAC 제어기의 평가 (II)  
(b) Evaluation (II) of proposed CMAC controller

그림 8 제안한 CMAC 신경망 제어기의 평가  
Fig. 8 Evaluations of proposed CMAC neural network controller

지연 소자를 이용하는 방법은 실험적인 방법으로 Dead-Zone을 구할 필요가 없고 하드웨어적인 구현이 간단하기 때문에 시스템의 발산을 막고 성능을 개선시키기 위한 효과적인 방법으로 생각된다. CMAC 제어기의 성능을 평가하기 위하여 직류전동기의 제어를 모의 실험하였다. 일반적인 직류전동기의 전달함수는 다음과 같이 표현된다.

$$\frac{\theta(s)}{V(s)} \cong \frac{1}{s} \left[ \frac{K_t / (RB_m)}{(1 + T_m s) + K_b K_t / (RB_m)} \right] \quad (8)$$

그리고 모의 실험을 위하여 직류전동기의 사양을 표 2와 같이 선정하였다.

표 2 직류전동기 사양  
Table 2 Specifications of DC motor

정격 전압	75 [V]
정격 출력	200 [W]
역기전력 상수(K <sub>b</sub> )	2.25 × 10 <sup>-2</sup> [V/rpm]
토크 정수(K <sub>t</sub> )	2.2 × 10 <sup>-2</sup> [N.m/A]
전기자 관성(J <sub>m</sub> )	2.47 × 10 <sup>-4</sup> [kgm <sup>2</sup> ]
전기자 저항(R)	2.38 [Ω]
기계적 시상수(T <sub>m</sub> )	12.68 [ms]
전기적 시상수(T <sub>e</sub> )	1.6 [ms]
접성 마찰 계수(B <sub>m</sub> )	1.95 × 10 <sup>-2</sup> [N.s/m]

표 2의 파라미터를 이용하여 식 (8)을 간략화 하면 다음과 같은 전달함수를 구할 수 있다.

$$\frac{\theta(s)}{V(s)} \cong \frac{37.38}{s^2 + 79.7s} \quad (9)$$

컴퓨터에서 모의 실험을 하려면 식 (9)를 이산 시간 함수로 변환해야 한다. 이산 시간함수로 변환하기 위해 먼저 식 (10)과 같은 State-Space로 변환한다.

$$\begin{aligned} x' &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (10)$$

$$A = \begin{bmatrix} -79.7 & 0 \\ 1 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 37.38 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 \end{bmatrix}$$

식 (10)을 정리하면

$$\begin{aligned} x'_1 &= -79.7x_1 + u \\ x'_2 &= x_1 \\ y &= 37.38x_2 \end{aligned} \quad (11)$$

식 (11)를 이산 시간 전달 함수로 전환하면

$$y(t) = \frac{81.7}{80.7} y(t-1) - \frac{1}{80.7} y(t-2) + \frac{37.38}{80.7} u(t) \quad (12)$$

과 같은 직류전동기의 이산 시간 전달 함수를 얻을 수 있다.

CMAC 신경망 제어기를 구성하기 위하여 되먹임 제어기로써 PID 제어기를 사용하였으며, 반복적인 모의 실험을 통하여 우수한 특성을 지닌 PID 제어기의 이득값(K<sub>p</sub>=60, K<sub>i</sub>=1, K<sub>d</sub>=0.05)을 추출하여 사용하였다.

본 논문에서 제안한 신경망 제어기의 경우 PID 이득값을 반드시 최적으로 설정할 필요는 없지만 모의 실험 결과를 비교 위해 우수한 특성을 지닌 PID 이득값을 구하였다. 본 논문의 신경망 제어기의 경우 되먹임 제어기의 출력을 학습하기 때문에 되먹임 제어기의 성능이 우수할수록 신경망 제어기도 더욱 우수한 결과를 출력할 수 있다.

그림 9는 구형파의 입력을 주었을 때 제어기의 출력을 보여주고 있다.

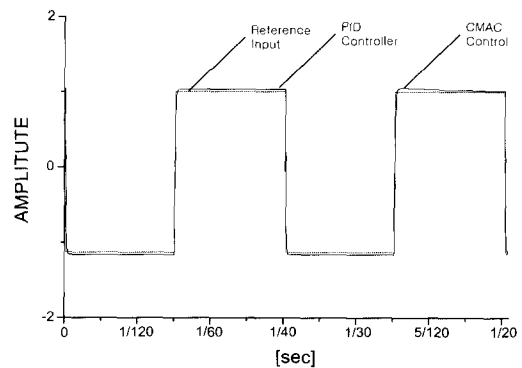


그림 9 구형파 입력에 대한 직류전동기의 속도응답  
Fig. 9 Speed response of DC motor for square wave

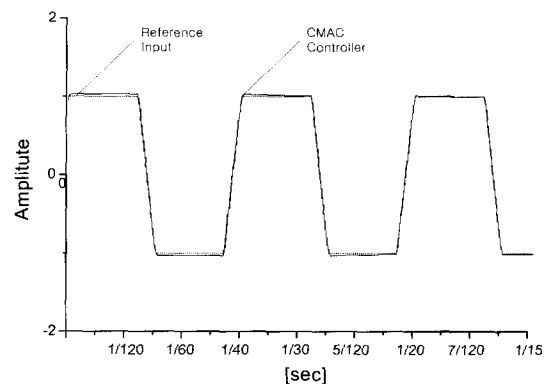


그림 10 사다리꼴파 입력에 대한 직류전동기의 속도응답  
Fig. 10 Speed responses of DC motor for trapezoid wave

모의 실험을 위해서 구형파의 기준 입력을 제어기에 입력하였다. 그림 9에서 나타나듯이 처음부터 1/27

[sec]까지는 PID 제어기만을 사용하였고 1/27 [sec]부터 CMAC 제어기를 동작시켰다. PID 제어기만을 사용하였을 경우 0.0306의 오차가 계속하여 발생하였으나 CMAC 제어기를 사용했을 경우 초기에 0.04의 오차를 보이다가 점차 감소하여 오차가 0으로 줄어드는 것을 확인할 수 있었다. CMAC 제어기의 초기 오차가 큰 이유는 입력이 갑자기 변화하게 될 경우 제너레이션의 효과가 없어지기 때문에 출력을 수렴하는데 있어서 초기에 많은 오차를 발생시킨다. 하지만 그림 10과 같이 입력이 연속적으로 변화할 경우 CMAC 제어기의 출력은 PID 제어기 보다 작은 초기오차를 보이며 결국 출력에 거의 정확하게 수렴하는 것을 알 수 있다.

### 5. 제안 제어기의 실시간 장력제어에 대한 적용 사례

제안 제어기의 성능 평가를 위해 웹 이송시스템을 구성하고 실시간 장력제어를 통해 검증한다. 웹 이송시스템은 구부러지기 쉬운 박판 또는 선을 재료로 하고 길이도 매우 길어 정확한 제어가 힘든 시스템이며, 특히 웹의 장력을 직접 제어할 수 없어 롤러의 모터 속도를 제어하여 웹의 장력을 조절하여야 하므로 제어에는 많은 변수가 작용할 수 있는 시스템이다. 제안 알고리즘의 검증을 위해 가장 간단한 전동기의 기준 속도 궤적에 대한 장력의 변화를 측정하는 것으로 CMAC 신경망 제어기는 직접 전동기의 속도를 제어하고 출력은 장력 센서를 통해 웹의 장력 상태를 측정하여, CMAC 신경망 제어기의 유용성을 시뮬레이션 결과와 비교하였다.

웹 재료는 중간제품 혹은 완제품을 제조하는 과정에서 일련의 연속적인 공정을 거치게 되며 각기 다른 공정을 거칠 때마다 장력의 수준 및 정확도가 다르게 된다. 장력의 변화는 공정 중에 재료의 파단, 제품의 품질 저하 등이 발생하므로 원하는 수준과 정확도를 유지하도록 제어하는 것이 매우 중요하다<sup>[10]</sup>. 이러한 장력제어는 최적 출력피드백 테크닉을 이용하여 구동 롤러의 속도를 제어하여 웹 스펀 양단에서의 속도차를 이용하여 웹의 장력을 제어하는 방법을 이용한다. 그러나 이런 방법은 장력 전달 현상에 의한 이웃하는 웹 스펀으로부터 외란의 영향을 제거할 수 없고 웹 스펀 간의 상호작용의 영향도 억제하기 어렵다. 또한 장력은 구동 롤러의 속도변화에 매우 민감하므로 롤러의 속도를 이용한 장력 제어에는 롤러속도의 극히 정확한 제어가 요구되어진다.

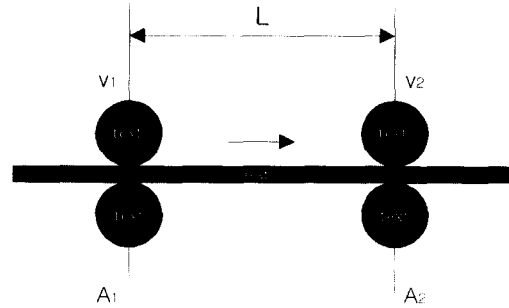


그림 11 단일 웹 스펀 시스템  
Fig. 11 Single web-span system

그림 11에 있는 단일 스펀 시스템을 이용하여 장력과 웹 속도의 관계는 식 (13)과 같이 나타낸다<sup>[10]</sup>. 이상된(idealized) 수학적 모델을 유도하기 위해서는 다음과 같은 조건을 가정한다.

- 1) 웹 재료와 롤러가 접촉하고 있는 구간의 웹 길이가 롤러 사이의 웹의 길이에 비해 무시할 만큼 짧다.
- 2) 웹의 두께는 롤러의 반경에 비해 매우 적다.
- 3) 웹과 롤러 사이에는 미끄러짐이 없다.
- 4) 단일 웹스팬 내에서는 온도나 습도 변화가 없다.
- 5) 웹의 단면적이 일정하며 웹의 변형도가 매우 적다.
- 6) 웹 스펀 내에서 변형도는 고르게 분포되어 있으며 웹 스펀 내에서 밀도와 탄성계수는 불변이다.

$$\frac{d}{dt}[T_2(t)] = -\frac{v_{20}}{L} T_2(t) + \frac{v_{10}}{L} T_1(t) + \frac{AE}{L} (V_2(t) - V_1(t)) \quad (13)$$

여기서  $T_n$ 은 정상상태 장력 값으로부터의 장력변화,  $v_{n0}$ 은 웹 속도의 정상상태 값,  $L$ 은 웹 스펀의 길이,  $A$ 는 웹의 단면적,  $E$ 는 웹의 탄성계수,  $V_n$ 은 웹 속도의 정상상태 값으로부터 변화를 나타낸다.

롤러는 웹 이송시스템에서 중요한 요소 중의 하나로 전동기에 의하여 구동되며, 그림 13과 같이 장력제어를 위한 시스템을 구성하고 제어기로는 CMAC 신경망 제어기를 이용하며 피드백 제어기로서 간단한 P 제어기를 사용한다. 웹 장력을 제어하기 위해 롤러를 구동하는 전동기의 속도를 제어하여야 하므로 제어기의 출력을 롤러의 전동기에 연결하고 전동기의 구동에 의해 발생하는 장력을 측정하기 위해 웹에 장력센서를

설치하였다. CMAC 신경망 제어기는 P 제어기의 파라미터를 반드시 최적으로 설정할 필요가 없으나 표 3과 같은 제어 파라미터를 사용하였고, 전동기 구동을 위해 그림 14와 같은 기준 속도 궤적을 사용하였다.

표 3 장력제어를 위한 제어 파라미터  
Fig. 3 Control parameters for tension control

Sampling Rate	1
Proportional Gain	1.0
Learning Rate	0.01
Quantization	199
Generation	9

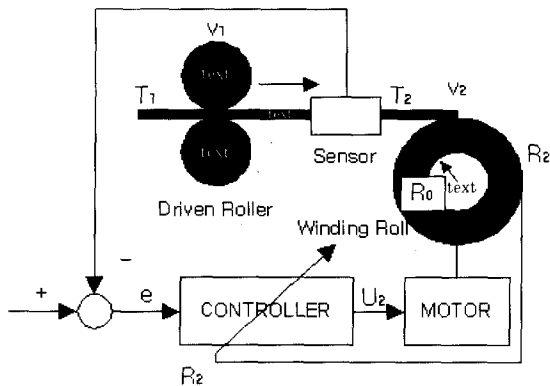


그림 13 장력제어기의 구성도  
Fig. 13 Block diagram of tension control system

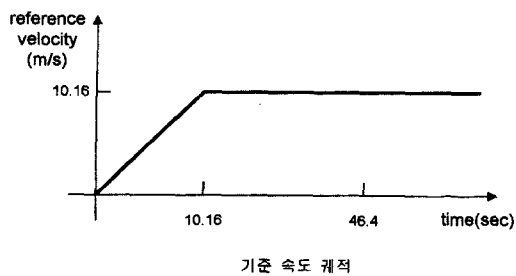


그림 14 기준 속도 궤적  
Fig. 14 Reference speed wave

그림 15는 롤러의 전동기 기준 속도 궤적에 대한 장력 센서의 출력으로 시뮬레이션 결과와 비교하였다. 제어 초기에는 기준 신호에 대해 시뮬레이션 결과와 CMAC 신경망 제어기 결과가 거의 일치하는 것으로 나타나고 있으나 점차 시간이 흐름에 따라 전동기의 속도와 웹 장력 센서의 출력간에 지연이 발생되어 오차가 누적되는 것으로 나타나 시뮬레이션 결과와 조금

차이를 보이고 있다. 이러한 결과는 웹 이송시스템에서의 장력제어는 구동 롤러의 전동기 속도를 제어하여야 하는데, 측정되는 장력 센서의 출력과 구동 롤러의 전동기 속도사이에 발생하는 지연시간에 의한 것으로 분석된다.

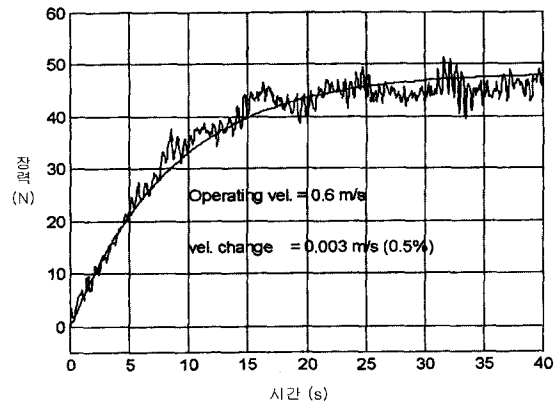


그림 15 CMAC 제어에 따른 장력 센서의 출력  
Fig. 15 Output of tension sensor by CMAC control

## 6. 결 론

본 논문에서는 CMAC 신경망을 사용하여 비선형 시스템을 식별하고 이를 제어하고자 한다. CMAC 신경망은 사람의 소뇌를 모방한 신경망으로서 복잡한 비선형 함수의 해를 수치적인 연산에 의해 구하지 않고 Table Look-up 방식을 이용하기 때문에 학습이 타 신경망에 비해 월등히 빠르고 용이하며 제어신호를 출력하기 위한 계산시간이 거의 필요치가 않기 때문에 비선형 시스템의 제어에 매우 유용하다. 본 논문에서는 CMAC 신경망의 이론적 배경에 대해 알아보고, 모의 실험을 통하여 간단한 비선형 함수에 대한 CMAC 제어기의 성능 증명을 하였다. 모의 실험 결과 제어기는 Tracking Error를 최소화하였으며, 실제 산업현장에서 사용하고 있는 직류전동기의 속도를 빠른 시간 내에 제어할 수 있음을 보였다. 특히 웹 이송시스템의 장력 제어에 CMAC 신경망 제어기를 이용하여 구동 롤러의 전동기 속도를 제어함으로써 시뮬레이션 값과 비슷한 장력 제어를 보임으로서 유용성을 입증하였다. 본 논문에서 제안한 CMAC 신경망 제어기를 사용할 경우 기존의 PID 제어기 대신에 사용할 수 있는 가능성을 시뮬레이션 및 실험을 통해 확인하였으며 IC화 할 경우 저렴한 비용의 인공 지능 제어기를 구성할 수 있을 것이다.



참 고 문 헌

- [1] E.K. Blum and L.K. Li, "Approximation Theory and Feedforward Networks," Neural Network, pp. 511~515, Apr. 1991.
- [2] W.T. Miller, F.H. Glanz and L.G. Kraft, "Application of a General Learning Algorithm to the Control of Robotic Manipulators," Int. J. Robotics Research, vol. 6, No. 2, pp. 84~98, Summer 1987.
- [3] W.T. Miller and R.P. Hewes, "Real time experiments in neural network-based learning control during high-speed nonrepetitive robotic operations," in Proc. 3rd Int. Svm. Intell. Contr., Arlington, VA, Aug. 24~26, 1988. pp. 513~518.
- [4] W.T. Miller, P.J. Latham, and S.M. Scalera, "Bipedal gait adaption for walking with dynamic balance," in Proc. Amer. Contr. Conf., Green Valley, AZ. vol. 2, 1991. pp. 1603~1608.
- [5] W.T. Miller, "Real-time neural network control of a biped walking robot," IEEE Contr. Syst. Mag, vol. 14, no. 1, pp. 41~48, Feb. 1994.
- [6] V. Donescu, D.O. Neacsu and G. Griva, "A Sysmetic Design Method for Fuzzy Logic Speed Controller for Brushless DC Motor Drives," PESC'96, vol. 1, pp. 689~694, June 1996.
- [7] 김형석, 최종수, "CMAC 신경회로망 기반 로봇 제어 기술," 전자공학회지 vol. 23, no. 12, pp. 136~144, Dec 1996.
- [8] Fu-Chuang Chen and Chih-Horng Chang, "Practical Stability Issues in CMAC Neural Network Control Systems," IEEE Trans. Control Systems. Vol. 4. No. 1. January 1996.
- [9] Chun-Shin Lin and Ching-Tsan Chang "Learning Convergence of CMAC Technique," IEEE Trans. Neural Networks. vol. 8. no. 6. Nov. 1997.
- [10] 신기현, "연속공정 시스템에서의 장력의 가변이득 PID 제어," 대한기계학회논문집 제 19권 제 3호, pp. 724~730, 1995.

저 자 소 개



**최광호(崔光皓)**

1973년 10월생. 1997년 건국대 전자공학과 졸업. 1999년 2월 동 대학원 졸업(석사). 현재 건아정보기술 연구원.



**조용범(趙鏞範)**

1959년 5월생. 1981년 경북대 전자공학과 졸업. 1988년 University of South Carolina 졸업(석사). 1992년 Case Western Reserve Univ. 졸업(박사). 1990년~1992년 J.W. Harley Inc. (GE Reuter-Stokes) 선임연구원. 현재 건국대학교 전자공학부 부교수.