

CMAC 디코더를 이용한 도립 진자 시스템의 안정화 제어에 관한 연구

정희원 박현규*, 이현도**, 한창훈***, 안기형*, 최부귀***

A Study on the Stabilization Control of an Inverted Pendulum System Using CMAC-Based Decoder

Hyun Kyu Park*, Hyun Do Lee**, Chang Hoon Han***,
Ki Hyung An*, Boo Kwi Choi*** *Regular Members*

요 약

본 논문은 Barto등에 의해 제안된 ASE와 ACE로 구성된 도립 진자 제어기에 CMAC 기반 디코더를 부가한 적응 평가 자기 학습 제어 시스템을 나타내었다. 시스템의 제어 목적은 가능한 한 오랫동안 진자의 균형을 유지하는 제어 출력을 생성하는 것이다. 그러나 제한되는 정보가 제한되어 있어 적응 평가기를 추가하여 다음 상태에 대한 예측 정보를 형성함으로써 이 문제를 해결하고, 또한 CMAC가 가지는 입력 공간의 일반화 성질을 이용하여 신경망의 학습수렴 속도를 증가시켰다. CMAC는 현재의 상태에 해당하는 정보와 그 상태에 대한 가능한 출력에 대한 정보를 ASE에 제공함으로써 ASE가 보다 더 정확한 출력을 생성하게 한다.

그리고 실제적인 문제에 대한 신경망 제어기의 적용성을 증명하기 위하여 실제 도립 진자 시스템을 제작하고, 신경망 제어기를 구성하여 실험을 행하였다.

ABSTRACT

This paper presents an adaptive critic self-learning control system with Cerebellar Model Articulation Controller (CMAC)-based decoder integrated with the Associative Search Element(ASE) and Adaptive Critic Element(ACE)-based scheme. The task of the system is to balance a pole that is hinged to a movable cart by applying forces to the cart's base. The problem is that error feedback information is limited. This problem can be solved when some adaptive control devices are involved. The ASE incorporates prediction information for reinforcement from a critic

* 부산정보대학 정보통신계열

** 통일중공업(주)

*** 동아대학교 전자공학과

論文番號 : 96009-0105

接受日字 : 1996年 1月 5日

to produce evaluative information for the plant. The CMAC-based decoder interprets one state to a set of pathways into the ASE/ACE. These signals correspond to the current state and its possible preceding action states. The CMAC's information interpolation improves the learning speed. And design inverted pendulum hardware system to show control capability with neural network.

I. 서 론

도립 진자 시스템은 여러 가지 다른 제어 알고리즘의 수행 능력을 비교하는 척도로서 사용되어지고 있다. 이것은 수직으로 세워진 진자의 균형을 유지하도록 학습하는 것을 제어 목적으로 한다.

현재의 신경회로망을 이용한 자기 학습 적응 제어의 빠른 발전 이전에는 도립 진자 문제에 있어 고전 제어 이론에 의한 광범위한 연구가 행하여졌고, 진자를 안정화시키는 많은 제어 시스템이 개발되었다.

Schaefer와 Cannon(1969)은 Bang-Bang 제어를 사용하여 진자를 안정화시키는 제어 시스템을 개발하였고,⁽¹⁾ Koeningsberg와 Frederick이 1970년 출력 제한 기법을 이용한 제어기를 제시하였다.⁽²⁾

그 이후 Strugeon과 Loscutoff(1972)는 상태 공간을 이용한 출력 제한 제어기로서 관측기 레귤레이터형 동적 안정화 제어기를 사용하여 도립진자를 안정화시킬 수 있는 제어 시스템을 설계하였다.⁽³⁾ 그리고 이러한 연구 결과를 근거로 Sehba(1983)와 Furuta(1984)는 이중 및 삼중 도립 진자 시스템에 대하여 컴퓨터 제어를 행하였으며⁽⁴⁾, Feng(1988) 등은 극점 배치법과 최적 제어법을 비교 분석하여 도립 진자 시스템의 안정화에 대한 시뮬레이션 결과를 발표하였다.⁽⁵⁾

이러한 모든 고전 제어 이론들은 시스템의 정확한 수학적 기술을 필요로 하며, 시스템의 물리적인 파라미터나 동역학적 특성에 대한 정보가 부족할 경우 고전 제어 이론을 사용한 분석 방법은 적용시키기 힘들어진다.

따라서 수학적 모델링과 해석보다는 학습 능력을 가지고서 임의의 입출력 사이의 비선형 함수 관계를 구현할 수 있는 인공 신경망 이론을 응용한 제어기가 시변 환경 하에서 더욱 강한 제어를 실현할 수 있다.

신경망에 의한 도립 진자의 안정화 제어에 관한 연구로는 1987년 Bernard Widrow가 Adaline으로 도립진자를 제어했으며, 1988년에는 진자의 상태를 센서 대신 시각 입력으로 대체하여 제어를 하였고, 같은 해에

Allon Guez는 프로그램 대신 훈련을 통해서 많은 설계가 on-line으로 행해지는 제어기의 일종인 훈련 가능한 적응 제어기(TRCS)를 제안하였다.⁽⁶⁾⁽⁷⁾⁽⁸⁾

또한, Barto, Sutton, Anderson(1983)과 Anderson(1989), Werbos(1987) 등에 의해 제안된 적응 평가학습(Adaptive Critic Learning)⁽⁹⁾⁽¹¹⁾은 경험에 의한 제어 동작 뿐 만 아니라 수행 능력 평가를 학습하는 인간과 유사한 자기 학습 방법이다.

본 논문에서는 Barto의 ASE/ACE 모델⁽⁹⁾⁽¹¹⁾에 CMAC⁽¹²⁾⁽¹⁵⁾ 기반 디코더를 응용한 새로운 형태의 CMAC 제어기를 제안하여 학습의 수렴 속도와 제어 성능을 향상 시켰다.

이러한 향상은 더욱 많은 변수를 가지는 복잡한 제어 문제에 있어서 신경망 기반의 자기 학습 제어를 보다 효과적으로 만든다. 또한 기존의 신경회로망을 이용한 도립 진자 시스템 제어에 관한 연구들이 컴퓨터 시뮬레이션에 그친 것에 비하여 본 논문에서는 신경회로망의 학습과 실시간 처리를 위한 소프트웨어 툴을 개발하고, 실제 도립 진자 시스템의 설계 제작 및 신경회로망 제어기를 구성하여 도립진자 시스템의 안정화 제어 실험을 행하였다.

II. 신경회로망 제어기

1. 도립 진자 시스템

도립 진자 시스템의 구조는 그림 1에서 보여지는 바와 같이 수평 방향의 레일 상에서 움직이는 활차와 그 위의 회전축에 연결된 진자로 구성되어 있다. 활차는 레일 위를 자유로이 움직일 수 있고, 진자는 제어력이 없는 상태에서 항상 어느 방향으로든지 넘어지게 되어 있다.

제어기는 시스템의 동역학적 파라미터인 활차의 위치, 진자의 각도, 활차의 속도, 진자의 각속도를 입력으로 가지고, 이것에 의해 일정한 크기의 힘(F)을 활차에 가하여 진자를 안정화시키는데 이 힘의 방향을 결정하는 것이 제어기의 출력이 된다.

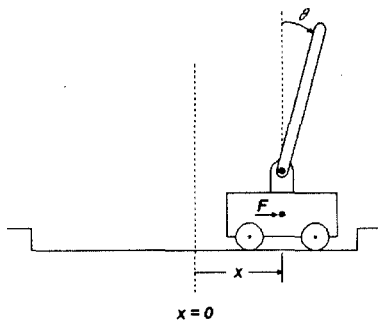


그림 1. 도립 진자 시스템
Fig. 1 Inverted pendulum system

1968년 Michie와 Chamber에 의해 연구되어진 이 시스템은 다음과 같은 4개의 상태 변수를 가진다.

- x : 레일상의 활차의 위치
- θ : 수직선상에서 기울어진 진자의 각도
- \dot{x} : 활차의 속도
- $\dot{\theta}$: 진자의 각속도

제어기는 매 시간 스텝마다 시스템의 상태를 나타내는 벡터를 입력으로 가지게 된다.

활차가 레일의 경계를 넘어서거나 진자가 넘어지면 실패 신호를 제어기에 제공하고, 시스템은 초기 상태로 되어 학습을 다시 시작하게 된다. 이 실패 신호 이외에 학습에 이용되는 제한 정보는 없다.

시스템의 제어 목적은 가능한 한 오랫동안 진자를 수직선상의 위치에 도립 시키기 위한 제어력을 생성하는 것이다.

시스템을 단순화하고, 선형화 하기 위해 1983년 Barto는 아래와 같이 시스템의 상태 공간을 162개의 박스(box)로 양자화 하였다.

- x : $\pm 0.8, \pm 2.4\text{m}$
- θ : $0, \pm 1, \pm 6, \pm 12^\circ$
- \dot{x} : $\pm 0.5, \pm \infty \text{ m/s}$
- $\dot{\theta}$: $\pm 50, \pm \infty \text{ }^\circ/\text{s}$

먼저 활차의 위치, 속도, 진자의 각속도는 3개의 레벨로 양자화하고 진자의 각도는 6개의 레벨로 양자화 한다. 그리고 양자화된 4개의 입력을 "4-162 디코더(decoder)"에 입력하여 출력 $x_i(t)$ 를 식 (2.1)에 의해 얻게 된다.

이것은 시스템의 가능한 모든 상태에 대하여 162개의 박스(box)를 생성하게 된다. 각 입력 상태에 따라 이들 162개 중 하나의 box만이 출력을 가지고 나머지는 0의 값을 가진다.

$$x_i(t) = \begin{cases} 1, & i = 54 \times x + 18 \times \dot{x} + 3 \times \theta + \dot{\theta} \\ 0, & \text{otherwise} \end{cases} \quad (2.1)$$

디코더는 시스템의 상태 벡터를 현재의 상태에 대응하는 하나의 box를 제외한 나머지는 0의 값을 가지는 2진(binary) 벡터로 변환하는 역할을 한다.

2. 강화 학습 제어기

1) ASE(Associative Search Element) 제어기
도립 진자 시스템은 다변수 비선형 시스템이기 때문에 시스템을 간략화하고 신경망의 학습을 쉽게 하기 위해서 입력으로 벡터 $x_i(t)$ 를 가지고 출력으로 $y(t)$ 를 가지는 단층 신경회로망을 사용하였다.

그림 2에 ASE(Adaptive Search Element)기반 학습 방법을 나타내었다.

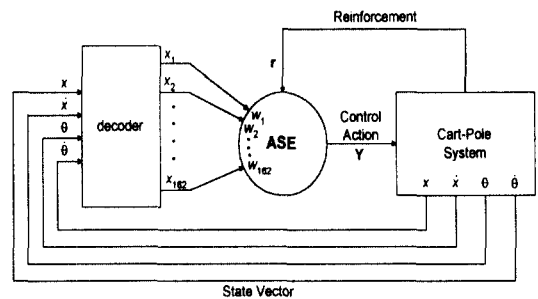


그림 2. ASE기반 제어기
Fig. 2 ASE based controller

시스템의 각 상태에 대한 입력 벡터를 $x_i(t)$ 로 놓고, ASE의 가중치 벡터를 $w_i(t)$ 로 놓는다. 그리고 자연적인 외란 성분을 포함시키면 입력 벡터에 의해 정의되는 출력 $y(t)$ 는 식 (2.2)와 같다.

$$y(t) = \sum w_i(t) \cdot x_i(t) + \text{noise}(t) \quad (2.2)$$

여기서 외란(noise)은 제로-가우시안(Zero-Gaussian) 평균을 가지는 랜덤 신호이다.

제어기의 출력은 제어력의 방향이 되고 전달 함수는 식 (2.3)으로 표시된다.

$$Y(t) = f[w_i(t) \cdot x_i(t) + noise(t)] \quad (2.3)$$

여기서 결정함수는 다음과 같다.

$$f[y(t)] = \begin{cases} +1, & y(t) \geq 0 \\ -1, & y(t) < 0 \end{cases} \quad (2.4)$$

2) ACE(Adaptive Critic Element) 제어기

ASE기반 신경망에 있어서 유일한 학습 정보는 시스템의 실패 신호와 연관된 강화 신호 r 만을 가지므로 실패 신호가 발생하기 전까지 ASE의 가중치는 변화되지 않고, 제어기는 실패 신호의 발생을 기다려야 하므로 학습의 수렴 속도가 느리다. 따라서 학습의 속도를 높이기 위하여 부가적인 학습 정보가 필요하게 된다.

파라미터 추정법에 있어 외부적인 학습기를 없애는 한 방법은 기존의 제어기를 복사함으로써 원하는 출력을 학습 정보로서 제공하는 것이다.

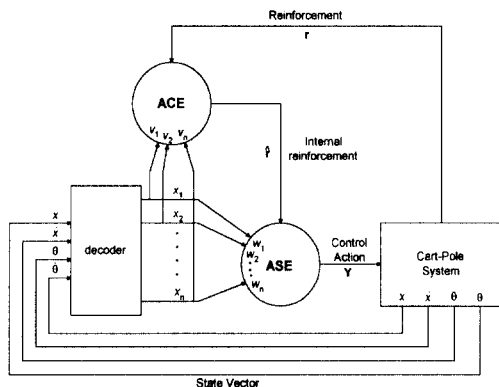


그림 3. ASE/ACE 신경망 제어기의 구조
Fig. 3 Structure of ASE/ACE neural network controller

이 이론에 기초하여 1983년 Barto는 단일 ASE에 단일 ACE(Adaptive Critic Element)를 부가한 새로운

형태의 도립 진자 제어기를 구성하였다. ASE/ACE 신경망 제어기의 구조를 그림 3에 제시하였다.

원하는 제어 목적을 수행하기 위한 기존의 제어기 ASE가 있고, 이것에 의해 신경망의 학습에 필요한 정보가 얻어진다. 주어진 입력에 대한 원하는 출력은 ASE에 의해 만들어진다.

ACE는 ASE와 동일한 입력을 가지고 ASE에 예측 정보를 제공하고 ASE는 ACE로부터 학습하게 되어 ACE의 구조를 복사하게 되고 ACE 또한 ASE의 구조를 복사한다. 결국 ACE는 ASE에 정확한 예측 정보를 제공하게 되고 ASE는 원하는 제어 출력을 생성하게 된다. ACE 알고리즘의 핵심은 예측 신호를 형성하고 현재의 강화 신호에 대한 예측뿐만 아니라 다음 스텝의 강화 신호에 대한 예측 능력을 가지게 된다.

ASE/ACE 제어기에서 $\hat{r}(t)$ 는 시간 스텝 t 에서의 ACE의 출력이고 이것은 ASE가 사용하는 개선된 강화 신호이다. 따라서 ASE 가중치 수정식에서 $r(t)$ 를 $\hat{r}(t)$ 로 바꾸면 개선된 ASE 가중치 수정식은 식 (2.7)과 같다.

$$w_i(t+1) = w_i(t) + a \hat{r}(t) e_i(t) \quad (2.7)$$

$\hat{r}(t)$ 를 생성하기 위하여 ACE는 먼저 다음 상태의 출력에 대한 예측 $\hat{p}(t)$ 를 결정한다.

$$\hat{p}(t) = \sum_{i=0}^{161} v_i(t) x_i(t) \quad (2.8)$$

여기서 $v_i(t)$ 는 ACE의 가중치 벡터이고 이것은 ASE의 가중치 벡터 $w_i(t)$ 와 유사한 역할을 하고 가중치 수정식도 ASE와 유사하다. ACE의 가중치 수정식을 식 (2.9)에 나타내었다.

$$v_i(t+1) = v_i(t) + \beta[r(t) + \gamma \hat{p}(t) - \hat{p}(t-1)]E_i(t) \quad (2.9)$$

여기서

- β ACE 가중치 변화율($\beta > 0$)
- γ 외부 강화 신호가 없을 때 예측 신호 감소율($0 < \gamma \leq 1$).

$v_i(t)$ ACE의 가중치 벡터

$\hat{r}(t)$ 내부 강화 신호

$E_i(t)$ ACE의 eligibility가 된다.

3. CMAC 기반 디코더

앞 절에서 Barto에 의해 고안된 ASE와 ACE를 이용한 신경망 도립 진자 제어 시스템에 대하여 논의하였다. 시스템의 각 상태에 대하여 디코더(decoder)는 하나의 유효한 출력을 내게 되고 162개 중 하나의 경로만이 현재의 제어 출력에 대한 책임을 가지고 가중치의 변화를 가져온다. ASE는 이러한 디코더(decoder)의 출력으로 제어 신호의 룩업 테이블(Look-up table)을 참조하게 되고 ACE는 룩업 테이블에 대한 예상 강화 신호의 값을 형성하게 된다. 각 정보는 주어진 위치에 대한 가중치의 값에 따른 집합으로 저장된다.

이러한 지역화된 저장 방법의 연속 동안 제한된 경로에 대한 일반화는 일어나지 않는다. 도립 진자 시스템은 연속적이므로 주위의 경로에 대한 정보들이 유용하다면 학습의 수렴 속도는 빨라지게 된다.

따라서 입력 공간의 일반화 성질을 가지는 CMAC (Cerebellar Model Articulation Controller)의 기법을 ASE/ACE에 적용함으로써 학습 속도의 개선을 가져올 수 있다.

CMAC은 수학적 연산보다 테이블을 참조한다. 입력 벡터로부터의 궤환 정보는 적절한 출력 값이 저장된 메모리상의 위치를 어드레싱(addressing)하게 된다. 즉 각 어드레스는 물리적인 메모리 위치들로 구성되어 있다. CMAC 메모리 어드레싱 알고리즘은 필요한 데이터를 실제적인 크기의 물리적 메모리 공간에 저장하는 것을 가능하게 하는 방법으로 제어 함수의 연속적인 성질을 이용한다. 흩어진 영역 대신에 중첩된 영역 내의 변수 값의 합하는 함수로서 테이블(table)을 참조하게 된다.

도립 진자 시스템에서 입력 경로는 상태 공간 벡터 $[x, \dot{x}, \theta, \dot{\theta}]$ 로 정의되고, 각 경로들은 서로 이웃하는 경로와 메모리를 나누어 정보들을 저장한다. 예로 식 (2.1)에 의하여 입력 경로의 숫자가 76이라면 이 경로의 상태 공간 벡터는 $[1, 1, 1, 1]$ 이 되고 $(76 = 1 \times 54 + 1 \times 18 + 1 \times 3 + 1)$ 이것은 메모리셀 $[0, 1, 1, 1], [1, 0, 1, 1], [1, 1, 0, 1], [1, 1, 1, 0], [0, 0, 1, 1] \dots$ 와 정보를 일부 나누어 가진다. 이웃하는 경로는 유사한 동역학적 상태를 가지고 출력 또한 유사하므로 메모리를 공유하는데 이러한 특성에 의해 일반화가

생성된다. 일반화는 학습에 필요한 정보를 중첩시킴으로 입력되지 않은 경로에 대해서도 학습하게 된다.

CMAC 알고리즘을 구현하기 위해서는 입력을 양자화 하는 디코더 부분의 변화가 필요하다. 그림 4에 CMAC기반 ASE/ACE 신경망 제어를 나타낸다.

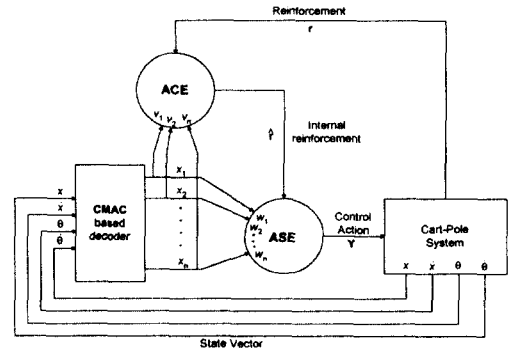


그림 4. CMAC기반 ASE/ACE 제어기
Fig. 4 CMAC based ASE/ACE controller

CMAC 디코더에 의해 162개 상태 중 하나와 이웃하는 상태의 경로가 ASE의 입력으로 디코딩 된다.

입력되는 상태는 이웃하는 상태와 유사한 정보를 나누어 가지지만 이웃하는 상태들 보다 제어 출력에 대한 책임은 더 크기 때문에 경로의 출력치가 다른 이웃하는 경로보다 더 큰 값을 가진다.

CMAC 디코더는 세 가지 서로 다른 레벨의 출력을 가지게 되는데 현재 상태에 대응하는 하나의 경로는 2의 크기를 가지고, 그 이웃하는 상태에 대한 경로들은 1의 크기를 가지고, 나머지 입력되지 않은 경로들은 0의 크기를 가지게 된다. 예를 들어 현재의 상태가 $[1, 1, 1, 1]$ 이라면 $x_{72}(t) = 2$ 이고, 이웃하는 상태 $[0, 1, 1, 1], [1, 0, 1, 1], [1, 1, 0, 1], [1, 1, 1, 0], [0, 0, 1, 1] \dots$ 은 $x_{22}(t) = x_{58}(t) = x_{73}(t) = x_{77}(t) = x_4(t) = \dots = 1$ 이 되고, 나머지 모든 $x_i(t) = 0$ 이 된다. 따라서 출력 $y(t)$ 는 다음과 같이 바뀌어 쓸 수 있다.

$$y(t) = \frac{\sum_{i=1}^{161} w_i(t) x_i(t)}{k} + noise(t) \quad (2.14)$$

여기서 k 는 디코더 출력의 수 즉, 현재 상태에 의해 입력되는 경로와 그 이웃하는 상태에 의한 경로의 합이 된다. $y(t)$ 의 크기가 입력의 수에 대하여 독립적이게 하기 위하여 가중치 합의 평균을 사용하였다.

이와 유사하게 ACE의 예측 함수도 다음과 같이 바꿀 수 있다.

$$p(t) = \frac{\sum_{i=0}^{k-1} v_i(t) x_i(t)}{k} \quad (2.15)$$

가중치 수정 함수, 적합 자취(eligibility trace) 함수, ASE와 ACE의 강화 신호 함수 등은 Barto의 모델과 동일하다. Barto의 모델에서는 학습 중의 매 시간 스텝마다 단지 하나의 경로만이 입력되는데 비하여 CMAC 기반 학습에서는 항상 여러 개의 경로가 입력되고 실패 신호가 없더라도 매 시간 스텝마다 가중치의 수정이 일어난다.

학습에 필요한 정보들이 각 상태에 대하여 중첩되어 있는 성질을 이용하므로 CMAC기반 제어기는 가중치의 수정에 있어 ASE/ACE기반 제어기보다 효율적이다. 또한 학습되지 않은 상태의 입력에 대하여 이미 학습된 상태와 유사한 출력을 생성하므로써 제어 문제에 대한 빠른 학습의 수렴을 보이게 된다.

III. 시뮬레이션

도립 진자 시스템의 수학적인 모델은 식 (3.1)과 식 (3.2)의 비선형 미분방정식으로 표현되어 진다.

$$\ddot{\theta}_t = \frac{g \sin \theta_t + \cos \theta_t \left[\frac{-F_t - ml \ddot{\theta}_t \sin \theta_t + \mu_t \operatorname{sgn}(\dot{x}_t)}{m_c + m} \right] - \frac{\mu_p \dot{\theta}_t}{ml}}{I \left[\frac{4}{3} - \frac{m \cos^2 \theta_t}{m_c + m} \right]} \quad (3.1)$$

$$\dot{x}_t = \frac{F_t + ml \left[\ddot{\theta}_t^2 \sin \theta_t - \ddot{\theta}_t \cos \theta_t \right] - \mu_c \operatorname{sgn}(\dot{x}_t)}{m_c + m} \quad (3.2)$$

시뮬레이션에서 사용되어진 시스템의 파라미터를 아래에 나타내었다.

$$g = -9.8 \text{ m/s}^2 \quad : \text{ 중력 가속도}$$

$$m_c = 1.0 \text{ kg} \quad : \text{ 활차의 질량}$$

$$m = 0.1 \text{ kg} \quad : \text{ 진자의 질량}$$

$$l = 0.5 \text{ m} \quad : \text{ 진자의 길이}$$

$$\mu_c = 0.0005 \quad : \text{ 활차의 마찰계수}$$

$$\mu_p = 0.000002 \quad : \text{ 진자의 마찰계수}$$

$$F_t = \pm 10 \text{ Newton} \quad : \text{ 시간 } t \text{에서 활차에 가해지는 제어력}$$

그리고 ASE/ACE 학습 파라미터는 다음과 같다.

$$\alpha = 1000$$

$$\beta = 0.5$$

$$\gamma = 0.9$$

$$\delta = 0.95$$

$$\lambda = 0.8$$

식 (3.1)과 식 (3.2)의 미분방정식을 Euler의 근사화 공식을 이용하면 아래와 같이 시간 스텝 Δt 에 대하여 나타낼 수 있다.

$$\dot{x}_{t+1} = \dot{x}_t + \ddot{x}_t \times \Delta t \quad (3.3)$$

$$\dot{\theta}_{t+1} = \dot{\theta}_t + \ddot{\theta}_t \times \Delta t \quad (3.4)$$

$$x_{t+1} = x_t + \dot{x}_t \times \Delta t \quad (3.5)$$

$$\theta_{t+1} = \theta_t + \dot{\theta}_t \times \Delta t \quad (3.6)$$

활차의 위치와 속도는 식 (3.3)과 식 (3.5)에 의하여 근사화 되지고 진자의 각도와 각속도는 식 (3.4)와 식 (3.6)에 의하여 근사화 되어진다.

식 (3.1)부터 식 (3.6)까지에 의하여 도립 진자 시스템의 상태를 근사화하고 시뮬레이션에 필요한 시스템의 상태 벡터를 얻을 수 있다.

시뮬레이션은 진자가 한계 각도($\pm 12^\circ$)를 넘어서거나 레일의 양쪽 끝($\pm 2.4\text{m}$)벗어날 경우 실패로 간주하며 실패 없이 60000회의 시간 스텝을 경과하면 성공으로 간주하였다. 시뮬레이션에서 시간 스텝(Δt)을 0.02초로 하였으므로 실제의 경우로 간주하면 20분에 해당하는 시간동안 진자의 균형을 유지할 경우이다. CMAC 기반 디코더 제어기가 기존의 ASE/ACE제어기 보다 학습 수렴 속도에 있어 향상됨을 보이기 위하여 똑같은 파라미터를 가지는 두 시스템을 학습시켜 각각 5번의 학습 결과를 그림 5와 그림 6에 나타내었다.

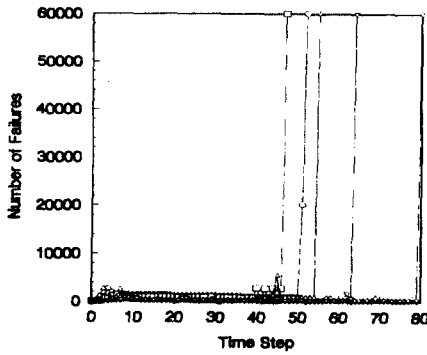


그림 5. ASE/ACE
Fig. 5 ASE/ACE

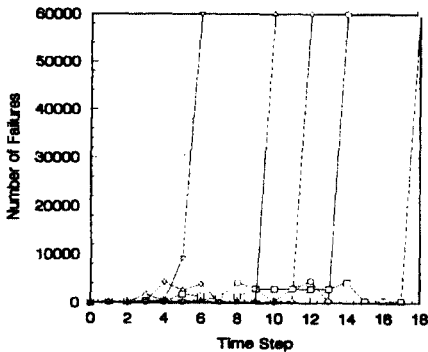


그림 6. CMAC
Fig. 6 CMAC

이상의 그림에서 보여지듯이 ASE/ACE 신경망에 의한 학습은 40회에서 80회 사이의 실패 과정을 거친 뒤 원하는 제어 목적인 60000 스텝의 시간 동안 균형 유지를 이루지만 CMAC 기반 디코더를 추가한 경우는 10회에서 20회 사이의 적은 실패로서도 원하는 제어 목적을 수행한다.

이것은 학습 중 입력된 상태와 이웃하는 상태들에 대해서 CMAC가 가지는 일반화 성질에 의하여 유사한 출력을 생성하므로 적은 양의 학습, 즉 적은 실패 횟수로도 원하는 제어 목적을 수행함을 의미한다.

따라서 시뮬레이션의 결과는 CMAC 기반 디코더를 추가함으로써 빠른 학습 수렴을 나타냄을 보여준다.

IV. 하드웨어 구현 및 실험

진자가 부착된 활차 부분은 길이 2m, 폭 20cm의 레일 위를 수평 방향으로만 이동하게 하였고, 이 레일은 이동 블록과 한 조로 되어 있어서 활차는 그 이동 블록의 위에 부착하였다.

활차의 질량은 그 탑재물을 포함하여 $M = 0.96\text{kg}$ 이다. 그리고 활차에 실리는 진자의 회전축에 로터리 엔코더를 부착하여 진자의 각도를 검출하였다. 진자는 질량 $m = 0.12\text{kg}$, 길이 50cm, 직경 6mm의 스텔레스 봉을 선택하였다.

또한, 활차를 움직이는 힘은 모터에 부착되어 있는 풀리에 감겨져 있는 와이어를 통하여 전달되고, 활차의 위치를 검출하기 위하여 모터 측에도 역시 로터리 엔코더가 부착되어 있다.

센서는 진자의 각도와 활차의 레일 상의 위치를 검출하기 위하여 2개가 필요하다. 각도 센서로서는 포텐셜미터나 로터리 엔코더가 사용되지만 포텐셜미터는 잡음의 영향을 받기가 쉽기 때문에 본 실험 장치에서는 광학식의 인클리멘탈형 로터리 엔코더를 사용하였다.

그리고 진자의 각도를 검출하는 로터리 엔코더는 1회전마다 2500개의 펄스를 발생하기 때문에 1펄스당 회전각은 $360^\circ/2500 = 0.144^\circ$ 로 된다.

와이어를 감은 풀리(외경 30mm, 내경 6mm)에 회전력을 주는 DC 서보 모터는 활차를 상당히 빨리 좌우로 구동시키지 않으면 안된다.

모터는 활차의 질량, 진자의 질량 및 길이 등을 고려하여 그것들을 움직이는데 필요한 토크를 가지는 DC 서보 모터를 선택을 하였다.

도립 진자 시스템을 제어하기 위한 시스템의 구성도를 그림 7에 나타내었다.

도립 진자 시스템의 안정화 제어 실험을 하기 위하여 2개의 위상 판별 회로와 타이머가 부착된 12bit의 up/down 카운터보드, 그리고 12bit의 D/A변환 보드 및 펜티엄 컴퓨터로 도립 진자 제어 시스템을 구성하였고, 신경망 알고리즘은 컴퓨터 상에서 프로그래밍 하였다.

그리고 사용한 D/A변환 보드는 분해능 12bit의 것으로서 1/4096의 분해능이 얻어진다. 이것은 구동장치의 출력 전압을 $\pm 10\text{V}$ 의 범위로 하면 1bit가 5mV 정도에 해당한다.

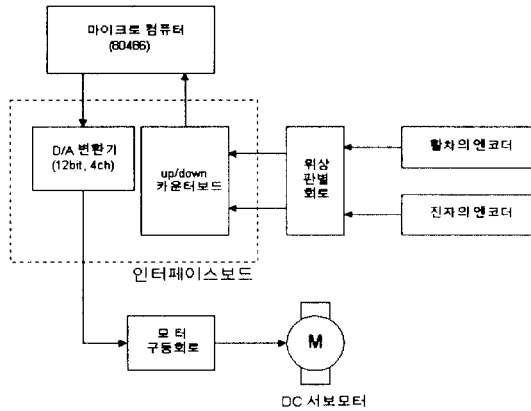


그림 7. 도립 진자 제어 시스템 구성도
Fig. 7 Block Diagram of the Inverted Pendulum System

실험에 앞서 충분한 시뮬레이션을 수행하여 CMAC 기반 디코더를 사용한 신경망이 학습의 수렴에 있어 우수함이 증명되었으므로 일반 디코더를 사용한 ASE/ACE 신경망 제어기의 실험은 생략하였고, CMAC 기반 디코더를 사용한 ASE/ACE 신경망 제어기를 프로그램 하여 실험을 행하였다. 그러므로 원래 목적하였던 도립 진자 시스템의 제어를 시스템에 대한 사전 정보 없이 신경망의 자기 학습 능력에 의하여 실제 시스템에 있어서도 원하는 제어 목적을 이룰 수 있는지를 보임으로써 실제 시스템에 대한 적용성을 보이고자 하였다.

이때 시뮬레이션에서와 같이 일정한 초기 각도를 주고 진자를 안정화시키는 방법은 실제 실험에 적용하기는 어려우므로 본 논문의 실험에서는 진자를 도립 상태에 위치시키고 어느 한 방향으로 10 Newton의 힘으로 움직인 다음 안정화 제어를 시작하는 방법을 사용하였다.

ASE/ACE 신경망의 파라미터는 시뮬레이션 설정과 같이 하였으며, 그 외 모든 시스템 파라미터도 시뮬레이션과 동일하게 설정하였다.

그림 8은 실제 시스템이 제어되는 과정을 나타내고 있으며, 그림 9와 그림 10은 각각 신경망이 학습 수렴된 이후에 활차의 위치와 진자의 각도에 대한 그래프를 나타내고 있다.

신경망 제어기의 학습이 수렴되었을 때는 진자의 움직임의 폭이 ± 0.1 rad 내외에서 좌우 대칭으로 계속 움직이고 있으며 활차의 위치는 느린 속도로 원점

주위를 이동하고 있다. 이와 같이 실제 시스템의 제어 성능도 시뮬레이션 결과와 유사함을 보였다. 이것은 좀더 복잡한 시스템의 제어 문제에 있어서도 신경망 제어기의 적용 가능성을 입증한다 할 수 있겠다.

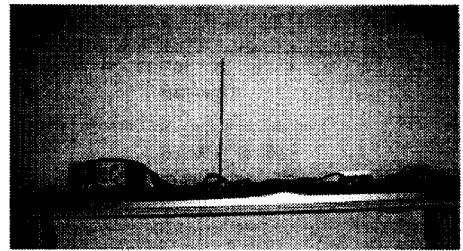
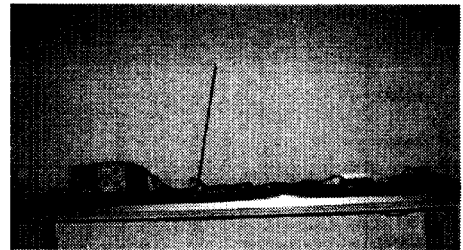
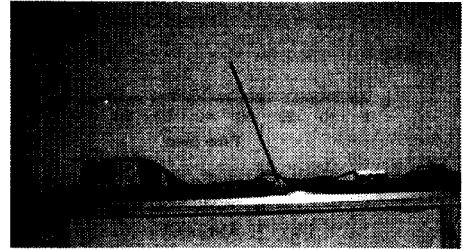


그림 8. 도립 진자 제어 과정

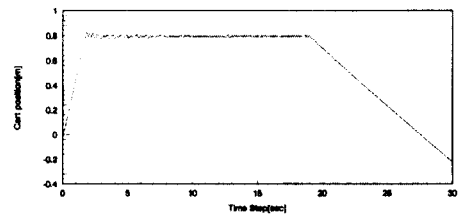


그림 9. 활차의 위치
Fig. 9 Position of the Cart

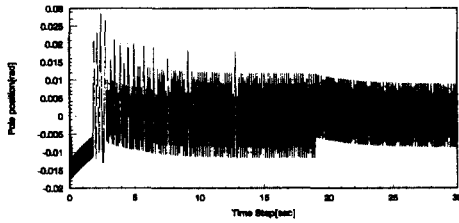


그림 10. 진자의 각도
Fig. 10 Angle of the Inverted Pendulum

V. 결 론

도립진자는 현대제어에서 불안정한 다변수 시스템의 전형적인 예로 논의되며 이는 2족보행 로봇, 로켓 발사 자세제어에 필요한 동역학의 기초가 된다. 따라서 새 제어이론 평가와 기존의 제어이론을 검증하는 척도로 사용되고 있다.

본 논문은 고전 제어이론에서는 진자의 안정화를 위하여 제어 대상이 되는 시스템에 대하여 시스템의 동역학에 관한 정확한 모델과 표현식을 포함하는 많은 정보가 필요하였으나 신경회로망을 이용한 제어에서는 그러한 시스템에 대한 사전 정보 없이 원하는 목적의 제어를 학습에 의해 수행할 수 있음을 보여주었고, 기존의 제어기에 새로운 형태의 CMAC 디코더를 응용함으로써 학습의 수렴속도를 개선시켰다. 그러므로 CMAC이 가지는 일반화 성질이 연속 시스템의 입력을 효율적으로 양자화 할 수 있음을 증명하였고 다변수 시스템의 제어문제에 대한 신경망 제어기의 적용성을 증명하였다.

앞으로의 연구에서는 좀더 나은 제어성능을 위하여 입력공간의 양자화 범위와 CMAC 알고리즘을 좀더 효율적으로 개선시킬 필요성이 있다고 생각된다.

참 고 문 헌

1. R.H. Cannon, "Dynamics of physical system", McGraw-hill, N.Y., 1967.
2. Katsuhiko Ogata, "System dynamics", Prentice-Hall, Inc., 1978.
3. Rolf Isermann, "Digital control systems", Springer-Verlag Berlin, Heidel., 1981.

4. K. Furuta and T. Okutani and H. Some, "Computer control of double inverted pendulum", Computer and Elec. Eng., Vol. 5, pp.67-84, 1978.
5. Qing Feng and Kazuo Yamafuki, "Design and simulation of control systems of an inverted pendulum", Robotica, Vol. 6, pp. 235-241, 1987.
6. W.T. Miller, "Sensor-based control of robotics manipulator using a general learning algorithm.", IEEE Journal of Robotics and Automat, RA-3, pp. 157-165, 1987.
7. C.W. Anderson, "Learning To Control An Inverted Pendulum Using Neural Networks.", IEEE Control Systems Magazine 9, pp. 31-37, 1989.
8. A.G. Barto, C.W. Anderson, R.S. Sutton, "Synthesis of Nonlinear Control Surfaces by a Layered Associative Search Network", Biology Cybernetics 43, pp. 175-185, 1982.
9. A.G. Barto, R.S. Sutton, C.W. Anderson, "Neuron-like Adaptive Elements That Can Solve Difficult Learning Control Problems.", IEEE Transactions on Systems, Man, and Cybernetics, SMX-13, pp. 834-846, 1983.
10. A.G. Barto, R.S. Sutton, "Landmark learning: An illustration of associative search.", Biology Cybernetics 42, pp. 1-8, 1981.
11. A.G. Barto, R.S. Sutton, "Associative Search Network: A Reinforcement Learning Associative Memory", Biology Cybernetics 40, pp. 201-211, 1981.
12. Albus, J.S. "A New Approach to Manipulator control: The Cerebellar Model Articulation Controller (CMAC)." Journal of Dynamics Systems, Measurement, and Control, pp. 220-227, 1975.
13. Albus, J.S. "Data Storage in the Cerebellar Model Articulation Controller(CMAC).", Journal of Dynamic Systems, Measurement and control, pp. 229-233, 1975.
14. D. Michie and R. A. Chambers, "BOXES : An experiment in adaptive control.", Machine Intelligence 2, pp. 137-152, 1968.
15. L.G. Kraft, D.P. Campagna, "A comparison between CMAC neural Network Control and two

