論 文
53D-5-7

자기 조직화 맵을 이용한 강화학습 제어기 설계

Design of Reinforcement Learning Controller with Self-Organizing Map

李 在 康* · 金 一 煥**
(Jae-Kang Lee · Il-Hwan Kim)

Abstract - This paper considers reinforcement learning control with the self-organizing map. Reinforcement learning uses the observable states of objective system and signals from interaction of the system and environment as input data. For fast learning in neural network training, it is necessary to reduce learning data. In this paper, we use the self-organizing map to partition the observable states. Partitioning states reduces the number of learning data which is used for training neural networks. And neural dynamic programming design method is used for the controller. For evaluating the designed reinforcement learning controller, an inverted pendulum on the cart system is simulated. The designed controller is composed of serial connection of self-organizing map and two Multi-layer Feed-Forward Neural Networks.

Key Words : Reinforcement Learning, Self-Organizing Map, Neural Dynamic Programming

1. 서 론

제어 대상에 대한 수학적 모델링을 이용하는 제어 방법에 있어서 사전에 정확한 수학적 모델을 구하는 것은 매우 힘들다. 또한 신경회로망을 이용하는 감독학습 방법에서는 현재 제어 대상의 상태에 대해 제어기가 수행해야 할 행동 데이터가 학습에 필요한데 모든 상태에 대한 제어기의 행동 데이터를 사전에 구하는 것은 거의 불가능하다. 이러한 문제를 해결하기 위해 제어 목표나 제어 범위와 같은 기본적인 사전정보만을 알고 있으면서 제어 대상과 환경 사이의 상호작용으로부터 학습이 이루어지는 신경회로망을 이용한 강화학습 제어 방법이 제안되었다[1]. 강화학습에서는 관찰 가능한 제어 대상의 상태와 강화 신호가 신경회로망 제어기의 학습에 사용된다. 강화 신호는 제어 대상과 환경간의 상호 작용으로부터 얻어지는데, 현재 상태에 대해 제어기가 취한 행동에 대한 평가를 이용한다. 일반적으로 성공 또는 실패의 두 가지 평가를 이용하므로 두 개의 스칼라 값을 이용해서 간단히 표현할 수 있다. 반면에 실제 시스템에서 관찰 되는 제어 대상의 상태는 연속신호이기 때문에 이론적으로 무한의 상태 수를 가질 수 있다. 이 연속신호를 디지털 컴퓨터를 이용한 실제 제어에 적용하기 위해서는 샘플링을 통해 이산화 해야 한다. 이산화 과정에서 정보의 손실 없이 정확한 시스템의 상태를 나타내기 위해 짧은 샘플링 주기를 이용한다고 하면 이산화된 상태는 그 수가 많아지게 된다. 이것을 그대로 신경회로망 제어기의 학습에 사용할 경우 정밀한 학습은 가능하지만 실제 시스템에 적용하기에는 많은 학습 시간이 소요되므로 적합하지 않다[2][3]. 그러므로 빠른 학습을 위해서는 학습이 가능한 범위에서 최소한의 상태의 수로 줄일 필요가 있다. 상태의 수를 줄이기 위한 방법으로 미리 지정한 값을 기준으로 상태를 분류하는 방법과 여기에 입력 공간 일반화 성질을 가지는 CMAC(Cerebellar Model Articulation Controller) 기법을 더불어 적용하는 방법이 제안되었다[4][5]. 하지만 이 방법들은 상태의 분포 특성을 유지하면서 분류할 수 있는 기준값을 정하기 위해서 기본적인 사전 정보 외에 부가적인 정보를 필요로 하는 문제가 있다. 한편, 상태의 수를 줄이기 위해 신경회로망의 일종인 자기 조직화 맵(Self-Organizing Map, SOM)을 이용하는 방법이 제안되었다[6][7]. 자기 조직화 맵은 경쟁학습을 통해 분포 특성을 유지하면서 상태의 수를 줄일 수 있는 방법이다[8]. 자기 조직화 맵은 신경회로망의 일종이므로 학습하는 동안에 부가적인 연산의 수행을 필요로 하게 되는데 이 부가적인 연산이 학습 속도에 미치는 영향을 줄이기 위해서 맵의 크기나 맵을 이용해 분류할 상태를 미리 알고 있는 기본적인 사전 정보를 토대로 결정할 수 있다.

강화학습 제어기는 제어기의 행동에 대한 평가를 하는 평가 네트워크와 평가를 토대로 주어진 상태에 대해 행동을 취하는 동작 네트워크로 구성 된다. 앞서 말한 두 가지 데이터를 토대로 크게 Temporal Difference(TD)학습 방법과 Heuristic dynamic programming(HDP) 학습 방법을 통해 학습이 이루어지게 되는데, 두 방법은 모두 평가 네트워크로부터 연속되는 두 평가 추정치 간의 temporal difference를 역전파 시킴으로써 미래의 동작 네트워크의 동작 평가가 최적

* 正 會 員 : 江原大學交 制御計測工學科 博士課程
** 正 會 員 : 江原大學交 電氣電子情報通信工學部 副教授 · 工博
接受日字 : 2003年 10月 6日
最終完了 : 2004年 3月 11日

화 되도록 한다는 기본 요소는 같다.

하지만 두 방법은 동작을 결정하는데 있어서 차이점이 있다. 먼저 TD 학습 방법은 동작 네트워크가 동작 정책을 학습함으로써 그 정책에 의해 동작이 결정된다[4][10]. 우선 평가 네트워크가 임의의 동작 정책으로 초기화 되서 동작을 취하게 된다. 그 동작 결과에 대한 평가를 강화 신호로 이용해서 평가 네트워크는 행동 네트워크가 따르는 행동 정책의 가치 평가를 TD 알고리즘으로 수행하고, 그 평가를 나타내는 값을 최대화 시키는 방향으로 동작 네트워크의 동작 정책을 갱신시키게 된다. 이 일련의 동작이 반복 되면서 동작 평가를 최대화 시키는 동작 정책이 동작 네트워크에서 학습 되게 된다. 이 방법은 매 번의 학습에서 동작 정책 전체가 갱신되므로 동작 네트워크의 초기 학습 시의 에러는 줄일 수 있지만 긴 학습 시간을 필요로 한다. 반면에 HDP 학습 방법은 동작 네트워크가 학습하는 것이 정책이 아닌 입력 상태들과에 그에 대한 동작을 연결지어 주는 방법을 사용한다[11]. 이 방법은 동작 네트워크가 전체 동작과 관련된 정책의 학습이 아닌 상태별로 취해야 할 동작의 연결을 학습하게 되므로 TD 학습 방법에 비해 학습 속도가 빠르다.

본 연구에서는 강화학습의 속도 향상을 위해 기본적으로 주어지는 사전 정보를 토대로 맵의 크기와 분류 대상 상태를 결정해서 자기 조직화 맵을 이용하여 신경회로망 제어기의 학습에 사용되는 상태의 수를 줄이고, HDP 학습 방법의 하나인 신경망 동적 프로그래밍(neural dynamic programming, NDP)을 통한 적응 평가 설계 (adaptive critic design) 방법을 이용하여 강화학습 제어기를 실제 시스템에 적용할 때 발생하는 온라인 학습시의 속도 문제를 해결하였다. 또한, 설계한 제어기를 도립진자 시스템에 적용하여 그 유용성을 확인하였다.

2. 제어기 구성

전체적인 제어시스템은 feed-forward neural network 형태의 동작 네트워크와 평가 네트워크로 구성된 신경회로망 제어기와 관찰 된 상태를 분류하는 자기 조직화 맵, 그리고 제어 대상의 연결로 이루어져 있다. 여기서는 자기 조직화 맵, 제어기의 학습 방법에 대해서 차례대로 설명하고자 한다.

2.1 자기 조직화 맵

제어 대상 시스템으로부터 관찰되는 상태는 연속 신호로부터 샘플링을 거쳐 얻어진 이산 신호이다. 이 이산 신호는 샘플링 주기에 따라 그 수가 많게도 적게도 될 수가 있다. 빠른 샘플링을 하게 되면 보다 정확한 상태 정보를 구할 수 있고 이것을 이용해서 학습을 하는 신경회로망은 정밀한 학습이 가능해 진다. 하지만 많은 데이터를 이용하기 때문에 학습 속도는 느려지게 된다. 반면에 느린 샘플링을 하게 되면 상태 정보가 부정확해 지고 이것을 이용해서 학습할 경우는 학습 성공의 경우는 빠른 학습이 가능하지만 부정확한 정보로 인해 학습의 성공률이 떨어지게 된다. 따라서 보다 정확한 상태 정보를 구해서 그 상태 정보의 특성을 유지하면서 학습에 사용되는 데이터의 수를 줄일 수 있다면 효율적인 학습이 이루어질 수 있게 된다. 본 연구에서는 이를 위해서 자기 조직화 맵을 사용하였다. 자기 조직화 맵은 신경회로망의 한 형태로 격자 상에 배치된 유닛의 형태로 구성이 되어 있다. 각각의 유닛은 무작위로 격자 상에 위치 한 상태로 초기화 되고, 학습 대상이 되는 데이터에 대해서 경쟁 학습을 통해서 그 데이터의 특성을 유지하면서 격자 상에 재배치된다. 이 때 재배치된 유닛의 가중치 벡터가 분류된 입력 데이터를 의미한다. 자기 조직화 맵의 학습 규칙을 간단히 살펴보면, 먼저 초기화 된 자기 조직화 맵에 입력 데이터가 주어지면, 각 유닛들과 입력 데이터 간의 거리를 계산한다. 이 계산 결과가 가장 작은 유닛, 즉 입력데이터와 가장 가까운 유닛이 입력을 가장 근사하게 나타나게 되므로 승자(winner)로 간주되어 그 유닛의 가중치가 입력 데이터를 향해 갱신 되게 된다. 자기 조직화 맵 상의 유닛을 t라고 하고, 입력 벡터를 $x=[x_1, x_2, \cdots, x_D]$, 그 유닛 t의 가중치 벡터는 $w^t=[w_1^t, w_2^t \cdots, w_D^t]$, 그리고 D는 입력 데이터의 차원 라고 하면 입력에 대한 유닛의 거리는 다음과 같이 계산된다.

$$\text{유닛과 입력 간의 거리} = \sum_{d=1}^{D}(x_d - w_d^t)^2 \quad (1)$$

이 계산 결과 거리가 가장 가까운 승자(winner)로 결정된 유닛의 가중치는 다음의 방법에 의해 갱신된다.

$$w^{winner} = w^{winner} + \beta(x - w^{winner}) \quad (2)$$

여기서 β는 학습률(learning rate)이다. 아울러 승자 유닛 외에 이웃한 유닛들도 neighbourhood function에 의거해 승자 유닛과 같은 방향으로 가중치가 갱신되게 된다.

자기 조직화 맵을 적용하는데 있어서 맵의 차원과 맵 상의 유닛 수를 정하는 것은 매우 중요하다. 분류해야 할 대상이 2가지이면 2차원 3가지이면 3차원의 맵을 구성해야 한다. 하지만 4차원이면 2차원의 맵 2개를 이용할 수도 있는데 이것은 복잡한 다차원에서의 연산을 단순화 시킬 수 있다는 장점이 있다. 본 연구에서는 관찰 가능한 상태의 수와 기본적으로 알 수 있는 사전 정보인 제어 범위와 샘플링 주기를 이용해서 유닛의 수와 맵의 형태를 결정하였다. 분류할 대상의 선정에 있어서는 관찰 가능한 모든 상태 중에서 오차의 누적에 대해 민감한 상태는 분류를 통한 오차를 고려해서 제외하는 방법을 사용하였다. 그림 1은 자기 조직화 맵을 이용해 분류할 입력 데이터를 표시한 것으로 그림 2에 보인 초기화 된 자기 조직화 맵이 학습해야 할 대상이 된다. 그림 3은 경쟁학습을 통해 학습이 끝난 자기 조직화 맵의 형태이다. 처음에는 임의의 위치에 배치되어 있던 자기 조직화 맵 상의 유닛들이 입력 데이터의 분포 형태로 재배치가 이루어진 것을 확인 해 볼 수 있다.

2.2 신경망 동적 프로그래밍 (Neural Dynamic Programming)

Heuristic dynamic programming을 사용하는 강화학습에서 신경 회로망이 학습해야 하는 대상은 두 가지로 나누어진다. 하나는 평가 네트워크에서 학습해야 하는 것으로, 평가 네트워크가 제어기가 취한 동작에 대한 평가를 나타내는 함수를 근사화 하도록 네트워크의 가중치를 갱신하는 것이다. 다른

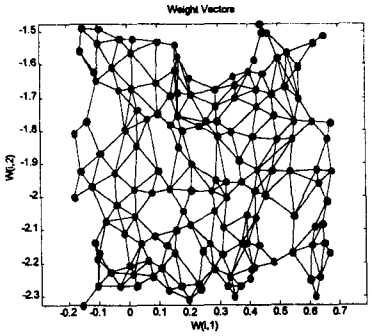

그림 1. 학습이 끝난 자기 조직화 맵
Fig. 1. SOM after learning

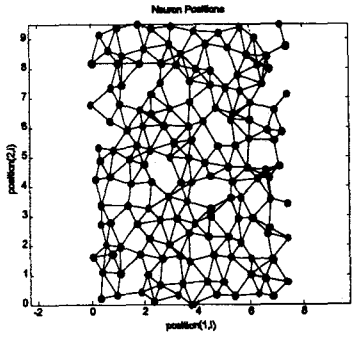

그림 2. 초기화 된 자기 조직화 맵
Fig. 2. Initialized SOM

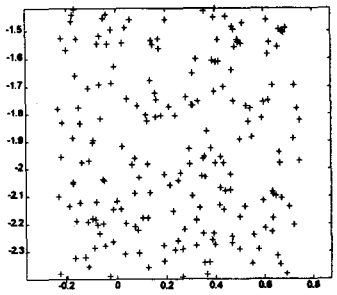

그림 3. 분류할 입력 데이터
Fig. 3. Input data

하나는 동작 네트워크에서 학습해야 하는 것으로, 제어기가 수행한 동작에 대한 평가가 최대화 되도록 현재 상태에 따라 어떤 동작을 수행해야 할지를 선택하도록 가중치를 갱신하는 것이다. 기본적으로 adaptive critic design에서 평가 네트워크가 학습해야 하는 함수는 Bellman 방정식을 만족하는 함수이다. 본 연구에서는 heuristic dynamic programming (HDP) 중에서 제어기에서 취한 동작이 평가 네트워크의 입력으로 사용되는 action dependant heuristic dynamic programming(ADHDP)방법을 기본으로 하는 neural dynamic programming(NDP) 학습 방법을 사용하였다. 대상 시스템의 모델을 사용하지 않는 점이 ADHDP와는 다르다. NDP에서는 동작 네트워크가 학습해야 할 함수는 다음의 방정식의 근사해를 구함으로써 얻어지게 된다.

$$J^*(X(t)) = \min_{u(t)} \{J^*(X(t+1)) + g(X(t), X(t+1)) - U_0\} \quad (3)$$

여기서 $g(X(t), X(t+1))$는 시간 t에서 $u(t)$에 의해 발생하는 비용을 의미하며, U_0는 양 변의 균형을 위해 추가된 heuristic 항이다. 평가 네트워크에 $J(X(t))$를 적용시키기 위해서는 식 (3)의 등호 오른쪽 부분을 미리 알고 있어야만 한다. 그러기 위해서는 다음 입력이 들어올 때 까지 기다리거나 시스템의 동적 특성을 미리 학습된 모델을 이용하는 방법들이 있다. 반면에 NDP에서는 이전 단계의 J는 물론 현재의 J를 저장함으로써 전체 네트워크의 연산량과 메모리 요구량이 늘어나기는 하지만, 다음 입력이 들어올 때 까지 기다리거나, 시스템의 동적 특성을 미리 학습한 모델이 필요하지 않게 된다. 그림 4는 제어대상 시스템을 포함한 전체 블록선도이다.

2.3 평가 네트워크와 동작 네트워크의 학습

평가 네트워크와 동작 네트워크는 모두 Multi-layer feed-forward network으로 구성되었다. 본 연구에서는 하나의 hidden layer와 하나의 output layer를 갖는 네트워크를 이용하였으며, 각 layer에서 사용한 전달 함수는 식 (4)와 같다.

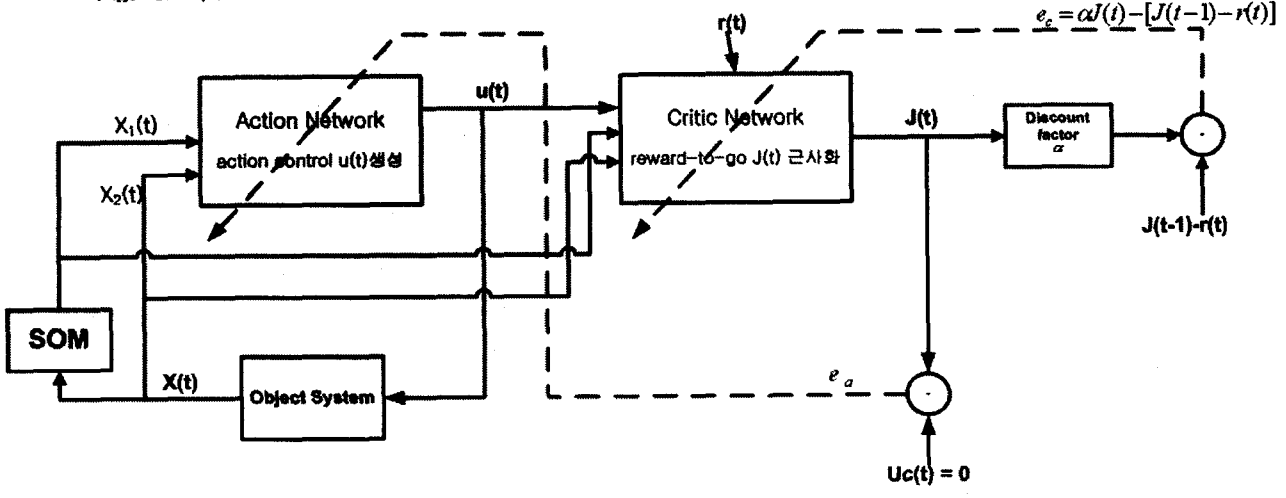

그림 4. 제어 시스템 전체의 블록선도
Fig. 4. Block diagram of control system

hidden layer 전달함수 : $y=\frac{1-e^{-x}}{1+e^{-x}}$

output layer 전달함수 : $y=x$ (4)

그림 5와 6은 각각 평가 네트워크와 동작 네트워크의 형태를 나타내는 블록선도이며, 동작 네트워크는 입력에 u가 포함되지 않았다는 점을 빼고는 평가 네트워크와 같은 형태로 구성된다. w_a^{hidden}, w_a^{output}, w_o^{hidden}, w_o^{output}은 각각 동작 네트워크와 평가 네트워크의 hidden layer와 output layer의 가중치를, $x_1, x_2, \cdots, x_n$은 관찰된 상태를, 그리고 u는 동작 네트워크의 출력을 의미하고 J는 평가 네트워크의 출력을 의미한다.

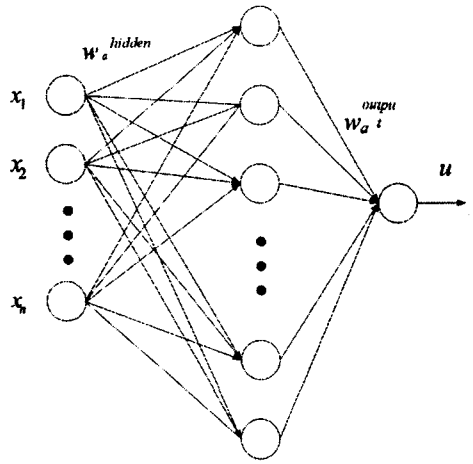

그림 5. 동작 네트워크 블록선도
Fig. 5. Action network

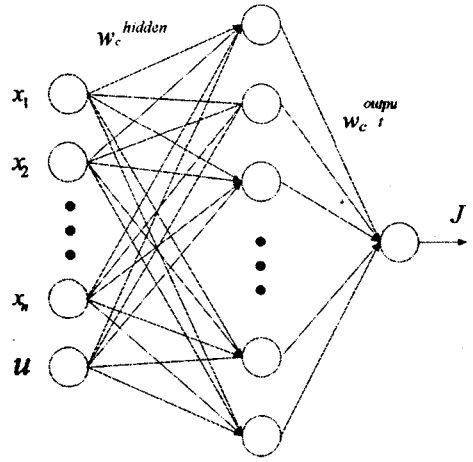

그림 6. 평가 네트워크 블록선도
Fig. 6. Evaluation network

평가 네트워크의 외부 강화신호인 $r(t)$는 제어기가 취한 동작이 성공일 경우는 0, 실패일 경우는 -1의 값을 가진다. 우선 두 네트워크의 학습 시작 시에는 모든 가중치가 랜덤하게 초기화 된다. 일단 현재 상태가 관찰되면 동작 네트워크에서는 현재의 가중치에 의해 동작을 취하게 된다. 주어진 상황에 대해 더 좋은 평가가 구해지는 동작을 취하게 되면 **최적화 원리**(Principle of optimality)에 의해 구해지는 최적의 동작 시퀀스에 좀 더 가깝게 된다. 최적의 동작 시퀀스는 다시 말해서 최대의 평가가 내려지는 동작 시퀀스이다. 따라서 동작 네트워크에서는 최적의 동작 시퀀스를 취할 수 있는 방향으로 가중치가 갱신되게 된다. 평가 네트워크에서는 상태와 동작 그리고 강화 신호를 토대로 동작에 대한 평가를 내리게 되는데 그림 4에서 평가네트워크의 출력 $J(t)$는 시간 t에서 앞으로 얻어질 모든 평가의 합을 나타낸다. 본 연구에서는 $J(t)$가 $R(t)$를 근사화 하도록 평가 네트워크의 가중치가 갱신 된다.

$$R(t)=r(t+1)+\alpha r(t+2)+\cdots \quad (5)$$

여기서 $R(t)$는 시간 t에서 discount된 앞으로의 평가의 합을 나타내며, α는 infinite-horizon 문제를 해결하기 위해 도입된 discount 인자이고 $(0<\alpha<1)$, $r(t+1)$은 시간 $t+1$에서의 외부 강화신호를 의미한다.

평가 네트워크와 동작 네트워크 모두 가중치 갱신에 경사하강 규칙(gradient descent rule)을 사용한다. 먼저 평가 네트워크에서의 예측 에러를 다음과 같이 정의 하면,

$$e_c(t)=\alpha J(t)-[J(t-1)-r(t)] \quad (6)$$

최소화해야 하는 목적 함수는

$$E_c(t)=\frac{1}{2}e_c^2(t) \quad (7)$$

가 된다. 따라서 가중치는 경사하강 규칙에 따라 다음과 같이 갱신 된다.

$$w_c(t+1)=w_c(t)+\Delta w_c(t)$$
$$\Delta w_c(t)=l_c(t)\left[-\frac{\partial E_c(t)}{\partial w_c(t)}\right] \quad (8)$$

여기서 $l_c(t)>0$ 은 시간 t에서의 학습률(learning rate)이고, w_c는 가중치 벡터이다.

동작 네트워크에서는 제어 목표인 U_c와 평가 네트워크에서 근사화된 $J(t)$의 차이를 통해 학습이 이루어진다. J(t)와 U_c의 차이를 다음과 같이 구성하면,

$$e_a(t)=J(t)-U_c \quad (9)$$

최소화해야 하는 목적 함수는

$$E_a(t)=\frac{1}{2}e_a^2(t) \quad (10)$$

가 된다. 역시 가중치의 갱신은 경사 하강 규칙에 따라 다음과 같이 갱신 된다.

$$w_a(t+1)=w_a(t)+\Delta w_a(t)$$
$$\Delta w_a(t)=l_a(t)\left[-\frac{\partial E_a(t)}{\partial w_a(t)}\right] \quad (11)$$

여기서 $l_a(t)>0$ 은 시간 t에서의 학습률(learning rate)이고, w_a는 가중치 벡터이다.

3. 모의실험

3.1 모의실험 구성

NDP와 SOM을 적용한 제어기를 도립진자의 자세 제어에

적용하여 모의실험을 통해 성능 검증을 하였다. 그림 7은 모의실험에 사용된 도립진자 시스템의 간략한 그림이고, 식 (12)는 이 도립진자 시스템의 운동방정식이다. 일반적으로 많이 사용되는 도립진자 시스템과 마찬가지로 제한된 트랙 위를 움직이는 수레 위에 막대가 연결되어 있는 형태로 목적은 막대가 수직을 유지하면서 수레가 트랙의 가운데에 위치하도록 하는 것이다. 이 운동 방정식은 TD 학습방법을 이용한 Charles W. Anderson이 사용한 것과 같은 방정식이다[2][3].

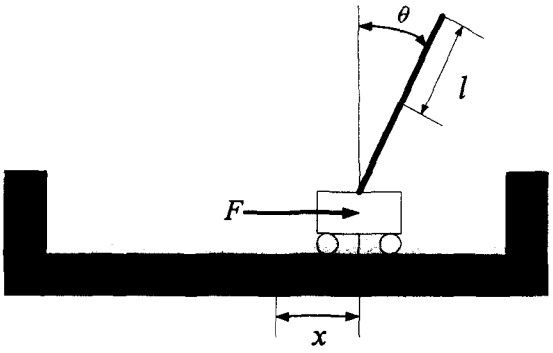

그림 7. 도립진자 시스템
Fig. 7. Inverted pendulum system

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

$$\frac{d^2x}{dt^2} = \frac{F + ml\left[\dot{\theta}^2\sin\theta - \ddot{\theta}\cos\theta\right] - \mu_c sgn(\dot{x})}{m_c+m} \quad (12)$$

표 1은 여기서 사용하는 도립진자 시스템의 파라미터들의 의미와 값을 나타낸다.

표 1. 도립진자 시스템의 파라미터
Table 1. Parameters of inverted pendulum system

파라미터	의 미	값
g	중력가속도	$9.8\ m/s^2$
m_c	수레의 질량	1.0 kg
l	막대 길이의 1/2	0.5m
μ_c	트랙 위의 수레의 마찰계수	0.0005
μ_p	수레 위의 막대의 마찰계수	0.000002
F	수레 질량 중심에 가해지는 힘	±10N

이 도립진자 시스템으로부터 수레의 위치 $x(t)$, 수직에 대해 막대가 기울어진 각도 $\theta(t)$, 수레의 속도 $\dot{x}(t)$, 그리고 막대의 각속도 $\dot{\theta}(x)$의 4개의 상태를 관찰할 수 있다. 수레가 움직이는 트랙의 범위는 [-2.4m, 2.4m] 이고, 막대가 [-12°, 12°] 범위를 넘어섰을 때를 쓰러진 것으로 간주하였다. 외부 강화 신호는 수레가 트랙 가장자리에 도달하거나 막대가 제어 범위를 벗어났을 때 실패 신호를 그 외에는 성공 신호를 발생시킨다. 수레에 가해지는 힘은 10N으로 고정되어 있으며 가해지는 방향만 바뀌게 되어 있다. 0.02sec를 시간 단위로 6000스텝, 즉 2분 동안 막대가 쓰러지지 않고 수레가 트랙 가장자리에 도달하지 않았을 때 학습이 성공한 것으로 정의 하였다. 관측 가능한 상태가 4가지이므로 자기 조직화 맵의 입력으로 위의 4가지 상태가 사용되어야 하는데 수레의 속도와 막대의 각속도의 경우 이산화하고 자기 조직화 맵을 이용해서 분류 하는데 있어서 발생하는 오차에 대해 민감하므로 속도, 각속도는 그대로 학습에 사용하였다. 따라서 수레의 위치와 막대의 기울어진 각도만을 자기 조직화 맵으로 분류하여 사용하였다. 자기 조직화 맵은 제어 가능 범위를 고려해서 10x15, 즉 150개의 유닛으로 구성하였으며, 이렇게 구성했을 때 각도의 평균 분류 오차는 0.02rad, 즉, 1.14°이다. 관측된 도립진자의 상태는 이 자기 조직화 맵을 통해 분류되어 동작 네트워크와 평가 네트워크의 입력으로 사용된다.

3.2 모의실험 결과

NDP만을 적용했을 때와 NDP에 SOM을 함께 적용했을 때에 대해 각각 모의실험을 실시하였으며 표 2와 표 3은 각각 10번의 학습 시도 후의 결과를 나타낸다.

표 2. NDP만을 이용한 모의실험 결과
Table 2. Simulation result of NDP only

학습 성공률	학습 성공까지 시도 횟수			
	최소	최대	평균	표준편차
100%	4	58	32.8	36.2

표 3. NDP와 SOM을 함께 이용한 모의실험 결과
Table 3. Simulation result of NDP with SOM

학습 성공률	학습 성공까지 시도 횟수			
	최소	최대	평균	표준편차
100%	8	48	29.4	21.4

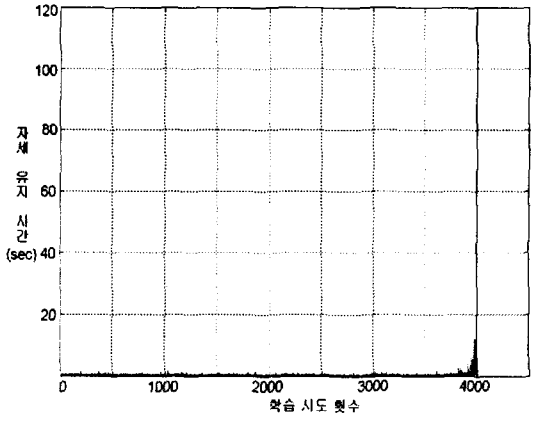

그림 8. 학습 성공까지의 자세 유지 시간
(TD 학습방법과 SOM 적용)
Fig. 8. Pole balancing time until success of learning
(TD learning with SOM)

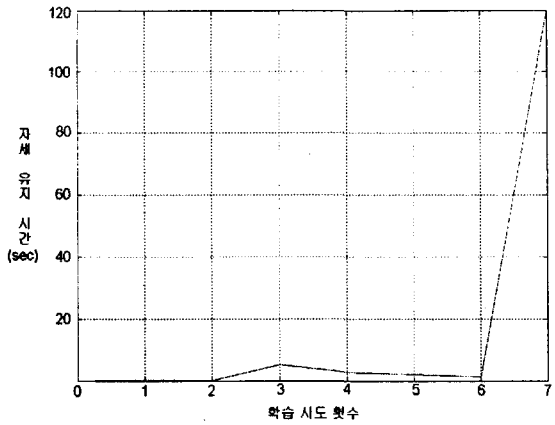

그림 9. 학습 성공까지의 자세 유지 시간 (NDP와 SOM 적용)

Fig. 9. Pole balancing time until success of learning (NDP with SOM)

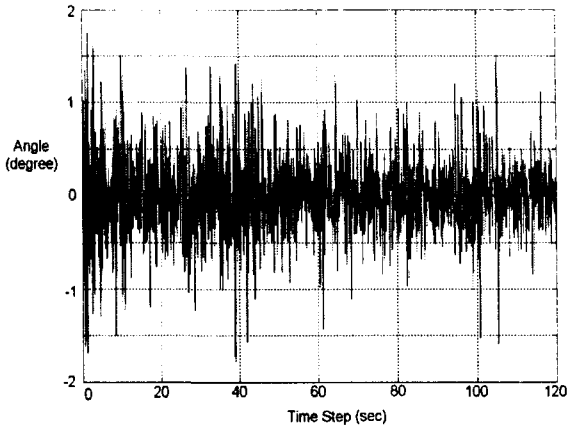

그림 10. TD와 SOM을 적용했을 때의 각도 변화

Fig. 10. Pole angle trace of TD with SOM case

TD 학습방법을 이용한 제어기의 10번의 평균 학습 성공까지의 시도 횟수는 3626번 이었다[2][3]. 여기에 SOM을 적용해서 같은 도립진자 시스템의 제어에 적용했을 때 10번의 평균 학습 성공까지의 시도 횟수는 2501번이었다. 그림 8은 TD 학습방법과 SOM을 이용한 모의실험의 결과에서 4004번 만에 학습이 성공했을 때의 매 학습 시도마다 얼마 동안 막대기의 자세를 유지 했는가를 나타낸 것이고 그림 9는 NDP와 SOM을 함께 적용한 모의실험 결과에서 7번 만에 학습이 성공했을 때의 유지 시간을 나타낸 것이다. TD 학습방법만을 이용했을 때에 비해 TD 학습방법에 SOM을 적용했을 때가, NDP만을 적용했을 때에 비해 NDP에 SOM을 적용했을 때가 학습 성공까지의 평균시도 횟수가 작다는 것을 확인 할 수 있다. 더불어 TD 학습 방법에 비해 NDP를 이용한 학습방법의 학습 속도가 월등히 빠른 것을 확인 할 수 있다. NDP만을 적용했을 때와 NDP와 SOM을 함께 적용했을 때의 결과를 비교해 보면 학습 성공 횟수의 표준 편차도 작게 나타난 것을 볼 수 있는데 이것은 학습 성공까지의 시간이 변화가 크지 않다는 것을 의미하며 이것은 언제나 비슷한 정도의 학습 능력을 보여준다는 점에서 제어기의 좋은 특성이라고 할 수 있다. 또한 TD 학습방법을 이용했을 경우 학습이 100% 성공하지 못했는데 반해 NDP와 SOM을 이용했을 경우에는 학습 성공률이 100% 이었다. 이것 또한 제어기의 좋은 특성이라고 할 수 있다.

그림 10과 11은 TD 학습방법과 SOM을 적용했을 때, 그림 12와 13은 NDP만을 적용했을 때, 그림 14와 15는 NDP와 SOM을 모두 적용했을 때의 막대기의 각도의 궤적과 수레의 위치를 나타내는 그래프이다. NDP를 이용하는 경우에는 모두 제어 목표에 만족하는 성능을 보여주는 것을 볼 수 있다. 하지만 TD 학습방법을 이용했을 경우에는 막대기의 각도를 수직으로 유지하고자 하는 제어 목표는 만족하지만 그림 11에서와 같이 수레의 위치를 트랙의 가운데로 위치시키는 제어 목표는 만족하지 못하는 경우도 발생하였다.

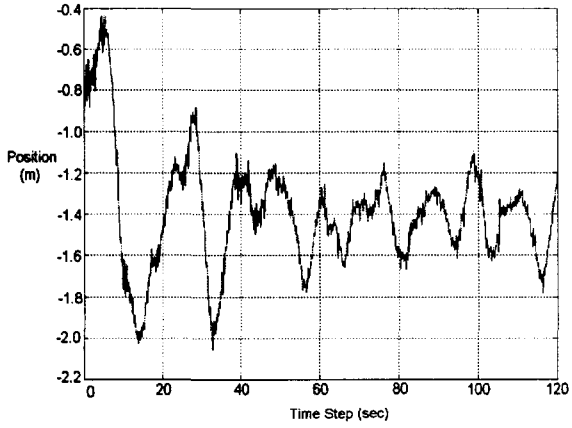

그림 11. TD와 SOM을 적용했을 때의 수레 위치 변화

Fig. 11. Cart position trace of TD with SOM case

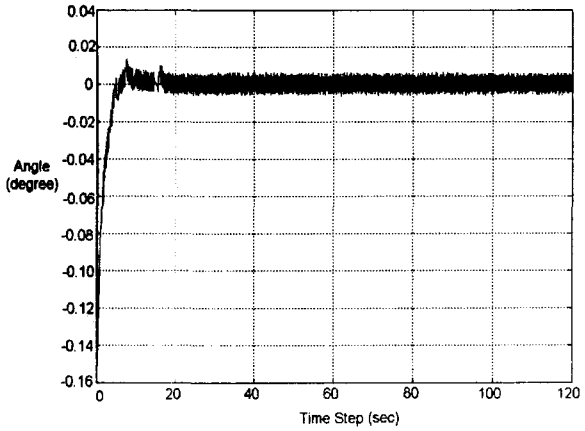

그림 12. NDP만을 이용했을 때의 각도 변화

Fig. 12. Pole angle trace of NDP only case

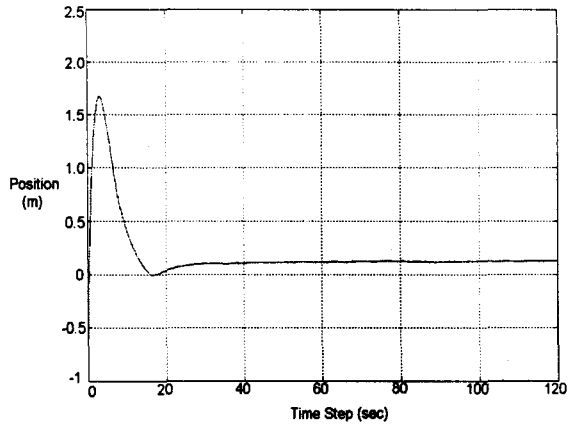

그림 13. NDP만을 이용했을 때의 수레위치 변화
Fig. 13. Cart position trace of NDP only case

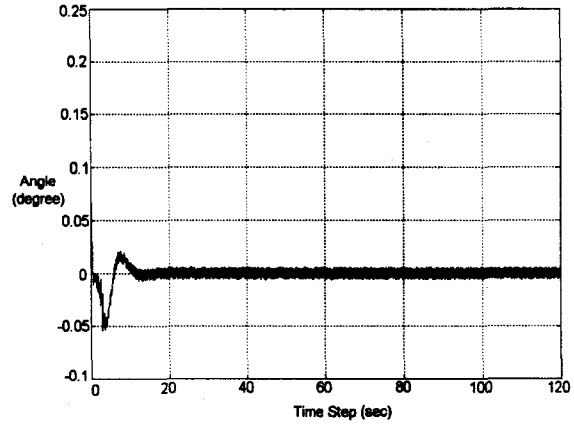

그림 14. NDP와 SOM을 이용했을 때의 각도 변화
Fig. 14. Pole angle trace of NDP with SOM case

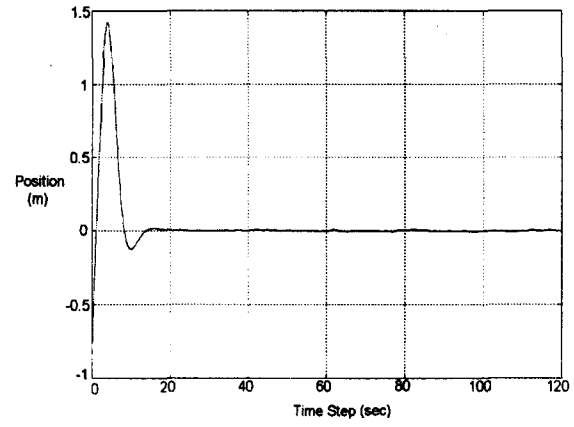

그림 15. NDP와 SOM을 이용했을 때의 수레위치 변화
Fig. 15. Cart position trace of NDP with SOM case

4. 결 론

기존의 강화학습에 대한 연구들의 대부분은 학습 성공에만 초점을 맞춰왔다. 하지만 강화학습 제어기를 실제 시스템에 적용하기 위해서는 빠른 학습 속도가 필요하다. 따라서 본 연구에서는 디지털 컴퓨터를 이용한 강화학습 제어에 있어서 학습 속도의 향상을 목표로 하였으며, 이를 위해서 heuristic dynamic programming의 하나인 neural dynamic programming을 학습 방법으로 사용하였고 자기 조직화 맵을 이용해서 데이터를 분류하였다. NDP를 이용하는 학습방법은 그 순간의 상태에 대한 적절한 동작을 연결시켜주는 방법을 이용한다는 점이 학습 속도를 빠르게 할 것이라는 점을 고려하여 선택하였고, 모의실험 결과를 통해서 확인할 수 있었다. 또한 강화학습에 사용되는 신경회로망 제어기의 학습 데이터를 줄임으로써 학습 속도를 향상 시키고자 하였다. 자기 조직화 맵은 격자상의 유닛의 배치를 입력 데이터를 이용한 경쟁학습을 통해 유닛을 재배치함으로써 데이터의 분포 특성을 유지하면서 데이터의 수를 줄일 수 있다는 특성이 있다는 점에 착안하여 제어기의 학습 데이터로 사용되는 상태의 수를 자기 조직화 맵을 이용하여 줄였다. 이 때 적용하는 자기 조직화 맵의 크기와 분류대상을 기본적으로 알 수 있는 사전정보인 제어범위와 샘플링 주기, 누적오차에 대한 민감도를 이용해서 결정하였으며, 그렇게 결정된 자기 조직화 맵을 적용함으로써 학습 속도가 향상된 것을 역시 모의실험 결과를 통해서 확인할 수 있었다. 아울러 자기 조직화 맵을 이용했을 때 학습 성공까지의 학습 시도 횟수 편차가 줄어든 것도 볼 수 있었는데, 이것은 안정적인 학습 성능을 보여줌으로써 제어기로서 좋은 특성이라고 할 수 있다.

본 연구에서는 강화학습 제어기의 학습속도 향상 방법을 제안하고, 모의실험을 통하여 제안한 방식의 유효성을 확인하였다.

감사의 글

이 논문은 2003년도 강원대학교 두뇌한국21사업 지원에 의하여 이루어진 연구로서, 관계부처에 감사를 드립니다.

참 고 문 헌

[1] Richard S. Sutton, and Andrew G. Barto, "Reinforcement Learning : An Introduction," MIT Press, Cambridge, MA, 1998.

[2] Charles W. Anderson, "Strategy Learning with Multilayer Connectionist Representations," Proceedings of the 4th International Workshop on Machine Learning, pp. 103-114, 1987.

[3] Charles W. Anderson, "Learning to Control an Inverted Pendulum Using Neural Network," IEEE Control Systems Magazine, Vol. 9, No. 3, pp. 31-37, 1989.

[4] Andrew G. Barto, Richard S. Sutton, and Charles W. Anderson, "Neuronlike Adaptive Elements That Can Solve Difficult Learning Control Problems," IEEE

Transactions on Systems, Man, and Cybernetics, Vol. SMC-13, No. 5, 1983.
[5] J. S. Albus, "A New Approach to Manipulator control : The Cerebellar Model Articulation Controller(CMAC)," Journal of Dynamics Systems, Measurement, and Control, pp. 220-227, 1975.
[6] Dean F. Hougen, Maria Gini, and James Slagle, "Partitioning input space for reinforcement learning for control," Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1917-1922, April, 1996.
[7] Andrew James Smith, "Applications of the self-organizing map to reinforcement learning," In Neural Networks (Special Issue), 15, pp. 1107-1124, 2002.
[8] T. Kohonen, "Self organizing maps," Berlin: Springer
[9] P. Werbos, "Advanced forecasting methods for global crisis warning and models of intelligence," General System Yearbook, Vol. 22, pp. 25-38, 1977.
[10] Richard S. Sutton, "Learning to predict by the methods of temporal difference," Machine Learning, Vol. 3, pp. 9-44, 1988.
[11] Jennie Si, and Yu-Tsung Wang, "On-Line Learning Control by Association and Reinforcement," IEEE Transactions on Neural Networks, Vol. 12, No. 2, pp. 264-276, 2001.

저 자 소 개

이 재 강(李 在 康)

강원대학교에서 제어계측 학사, 석사 학위를 각각 1997년과 1999년에 받았으며, 현재 동 대학원에서 박사과정 중에 있다. 관심 연구 분야는 인공 신경 회로망을 이용한 학습 제어이다.

김 일 환(金 一 煥)

서울대학교에서 제어계측 학사, 석사 학위를 각각 1982년과 1988년에 받았으며, 1993년에 일본 토호쿠대학에서 공학 박사 학위를 받았다. 1995년 강원대학교 전기전자정보통신공학부 교수로 임용되어 현재 동 학부 부교수로 재직 중이다. 관심 연구 분야는 제어, 메카트로닉스 및 휴먼 인터페이스이다.