산업기술연구(강원대학교 산업기술연구소 논문집), 제20권, 2000.
Journal of Industrial Technology, Kangwon Nat'l Univ.,Korea, No. 20, 2000.

신경회로망을 이용한 도립진자의 학습제어

Learning Control of Inverted Pendulum Using Neural Networks.

이 재 강* 김 일 환**
Lee, Jae-Kang Kim, Il-Hwan

Abstract

A priori information of object is needed to control in some well known control methods. But we can't always know a priori information of object in real world. In this paper, the inverted pendulum is simulated as a control task with the goal of learning to balance the pendulum with no a priori information using neural network controller. In contrast to other applications of neural networks to the inverted pendulum task, the performance feedback is unavailable on each training step, appearing only as a failure signal when the pendulum falls or reaches the bound of track. To solve this task, the delayed performance evaluation and the learning of nonlinear functions must be dealt. Reinforcement learning method is used for those issues.

키워드 : *강화학습, 신경회로망, 도립진자*
Keywords : Reinforcement Learning, Neural Network, Inverted Pendulum

1. 서론

도립진자의 자세 제어는 도립진자의 특성상 본질적으로 비안정이며 도립진자의 동역학 모델은 보행로봇이나 다른 좀더 복잡한 제어에 필요한 기본을 포함하고 있기 때문에 그동안 많은 제어이론의 적용대상이 되었다. 다른 여러 가지 제어이론을 통한 도립진자의 자세 제어에 있어서는 도립진자의 동역학 모델, 도립진자가 어떻게 동작해야 하는지에 대한 목표함수와 같은 도립진자에 대한 사전정보를 충분히 가지고 있어야 도립진자의 제어를 수행할 수 있었다. 하지만 실제로 제어이론을 적용하고자 할 때 언제나 대상에 대한 동역학 모델링이나 목표함수와 같은 상세한 사전정보를 알 수는 없다. 따라서 이러한 문제를 해결하고자 그동안 여러 제어방법이 제시되어 왔는데 그 중에서 Charles W. Anderson은 인공신경회로망 제어기의 강화학습을 통한 제어방법[1]을 제시하였다. 이 논문에서는 Charles W. Anderson이 제시한 방법을 토대로 이러한 강화학습을 통한 인공신경회로망을 이용한 도립진자의 자세 제어기를 구현해 보았으며 모의실험을 통해 적용해 보았다.

* 강원대학교 제어계측공학과 박사과정
** 강원대학교 제어계측공학과 교수, 공학박사

2. 강화학습의 기본개념

기본적인 강화학습 모델에서 학습대상은 학습대상의 환경과 상호작용을 하게 된다. 학습대상은 자신의 환경과의 상호작용의 결과를 입력받고 그 입력을 토대로 행동을 결정하게 되며, 그 행동으로 인해 어떤 방식으로의 환경의 변화를 일으키게 되

고, 그 변화가 강화학습에 대한 신호로 학습대상으로 전달돼 바람직한 상호작용 결과를 발생시키는 행동 방식을 학습하게 된다. 이러한 강화학습 모델을 이용하는 강화학습 문제에는 기본적인 구성 있는데 환경(environment), 강화함수(reinforcement function), 가치함수(value function)가 그것이다.

2.1 환경(The Environment)

모든 강화학습 시스템은 처해있는 상황에 대해 어떤 행동을 취해야 할지를 환경과 시스템간의 trial-and-error에 의한 상호작용결과를 통해 학습하게 된다. 따라서 환경은 강화학습 시스템에 의해 최소한 부분적으로라도 관찰 가능해야한다. 그리고 그 관찰 결과는 강화학습 시스템에 어떤 형태로 전달이 되어야 한다. 만일 강화학습 시스템이 환경에 대한 모든 정보를 완전히 관찰 가능하다면 강화학습 시스템은 확실한 환경의 상태를 토대로 행동을 선택할 수 있게 된다. 이러한 이상적인 환경은 강화학습에 있어서 가장 좋은 결과를 나타낼 수 있으며, 강화학습에 관련된 많은 이론들에 대해 필요조건이다.

2.2 강화함수(The Reinforcement Function)

앞의 환경에서 말했듯이 강화학습 시스템은 처해있는 상황에 대해 어떤 행동을 취해야 할지를 환경과 시스템간의 trial-and-error에 의한 상호작용결과를 통해 학습하게 된다. 이러한 강화학습 시스템에 있어서의 학습 목표는 강화함수의 개념을 통해 정의 될 수 있다. 각각의 상태에 대한 행동에 있어서 그 행동을 취함으로 인한 결과에 대한 보상이 어떤 스칼라 값(scalar value)으로 주어지게 되는데, 강화학습 시스템에서는 학습을 통해 행위자가 이 보상을 최대화하는 행동을 취할 수 있도록 하게 된다. 초기 상태로부터 최종 상태로 진행하는데 있어서의 이 보상 값을 나타내는 것이 바로 강화함수이다.

2.3 가치함수(The Value Function)

강화학습 시스템에서는 앞의 강화함수에서 말했듯이 보상을 최대화하는 행동이 필요하게 되는데 이제 상황에 대해 어떤 행동을 취하면 보상이 최대가 되는지를 어떻게 결정하는가 하는 문제가 남아있다. 이를 위해서 먼저 정책(policy)과 상황의 가치(value of a state)의 두 가지 요소를 정의하도록 한다. 정책(policy)은 각각의 상황에서 어떤 행동을 수행해야 할지를 결정한다. 다시 말해 각각의 상황을 수행해야 할 행동으로 대응시키는 역할을 한다. 상황의 가치는 그 상황에서 시작해서 어떤 정책(polity)을 따라서 최종 상황에 도달했을 때 받게 되는 보상의 합으로 정의한다. 따라서 최적의 정책(policy)은 임의의 시작상황으로부터 그 정책에 따라 행동을 대응시켜서 최종 상황에 도달했을 때 취한 행동에 대한 보상이 최대화 되는 정책을 말하게 된다. 위의 두 가지 정의로부터 상황의 가치는 정책에 의존한다는 것을 알 수 있다.

가치함수는 여러 가지 형태의 함수로 근사화 될 수 있는 상황(state)에 대해 상황의 가치를 대응시키는 함수를 의미한다.

3. 정책의 학습(Learning of Optimal policy)

강화학습을 통한 대상의 제어에 있어서 목표는 간단히 말해서 제어기가 최적의 정책을 학습하는 것 이라고 말할 수 있다. 강화학습을 이용한 제어 시스템에 있어서 정책의 학습에는 제어 대상의 모델 자체를 학습 하지 않고(model-free) 제어기를 학습시키는 방법과 우선 대상의 모델을 학습하고 그것을 토대로 제어기를 구성하는(Model-based) 방법이 있다. 이 논문에서는 AHC(Adaptive Heuristic Critic)과 TD(Temporal Difference) 방법을 이용하는 전자의 방법을 선택했다.

3.1 AHC와 TD(0)를 이용한 학습방법

AHC(Adaptive Heuristic Critic) 알고리즘은 가치 함수의 연산이 연립 선형방정식을 푸는 것이 것이 아니라 TD(0) 알고리즘에 의해 수행되는 것으로 정책 반복(policy iteration)[3]의 응용 형태이다.

그림 1은 AHC와 TD(0)를 이용한 강화학습 시스템의 블록선도 이다.

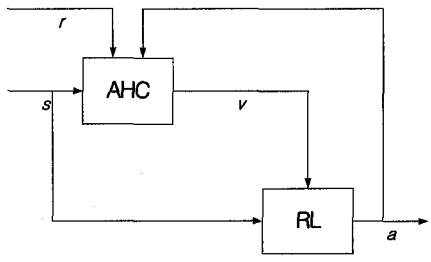

그림 1. 강화학습 시스템 구성

그림에서 RL은 강화학습 부분를 나타내고 있다. 강화학습 부분은 다중 상태를 처리할 수 있고 시간에 따라 변화하는 보상이 주어지는 k-armed bandit 알고리즘의 한가지 일 수 있다. AHC를 이

용한 강화학습 시스템에서의 강화학습 부분은 앞에서 말한 일반적인 행동에 대한 즉각적인 보상을 최대화하는 방향으로 행동을 취하는 것이 아니라 AHC의 평가를 나타내는 값인 v를 최대화 시키는 방향으로 행동을 한다. AHC는 실제 외부로 부터의 강화 신호(reinforcement signal)를 각각의 상태들에 대해 RL부분에서 현재 구체화되고 있는 정책에 의해 주어지는 기대값을 대응시키기 위해 학습에 사용한다. 그림에 나타나 있는 이 두 부분이 교대로 동작한다면 우리는 수정된 정책 반복(modified policy iteration)과 유사한점을 볼 수 있다. RL부분에 의해 수행되는 정책 π는 고정되고 그 정책에 대해 AHC는 가치 함수 $V\pi$를 학습한다. 다시 AHC를 고정시키고 RL부분이 새로운 가치함수를 최대화 하는 새로운 정책 π'을 학습하게 한다. 이렇게 계속 반복되는 것이다. 적절한 조건에서 오직 순차적인 수행만이 최적의 정책으로 수렴하는것을 보장할 수 있다. Williams 와Baird가 'incremental variants of policy iteration'이라고 불리는 AHC관련 알고리즘의 종류들의 수렴특성을 연구했다. [4]

이제 남은건 어떻게 AHC가 정책의 가치(value of a policy)를 학습하느냐 하는 것 이다. 실제 환경에서 상태 천이를 나타내는 $<s, a, r, s'>$을 정의하도록 하자. 여기서 s는 학습을 통해 실제로 동작을 하는 행위체의 천이 이전의 상태이고, a는 행위체의 행동의 선택, r은 행동에대해 받게 되는 즉각적인 보상, 그리고 s'는 행동으로 인한 결과 상태를 의미한다. 여기서 정책의 가치는 다음의 갱신 규칙을 통해서 Sutton이 제안한 TD(0) 알고리즘 [5]을 이용해서 학습된다.

$$V(s) = V(s) + \alpha(r + \gamma V(s') - V(s))$$

r은 즉각적인 보상을 나타내고 $V(s')$은 다음 상태에 실제로 발생하는 추정가치 이므로 상태 s가 되면 추정 가치는 $r+\gamma V(s')$에 가까워지도록 갱신된다. 이것은 정책 반복의 sample-backup 규칙과 유사하다. 단 한 가지 다른점은 sample이 알려진 모델의 모의 실험을 통해서가 아니라 실제 세계로부터 얻어진다는 점이다. 핵심 아이디어는 $r+\gamma V(s')$는 가치 $V(s)$의 샘플이라는 것이다. 만일 학습정도 α가 적절히 조절되고, 정책이 고정된다면, TD(0)는 최적의 가치함수로 수렴하는 것을 보장한다.

4. 모의실험

도립진자의 자세 제어는 비안정이면서 다중출력이 있는 동역학 시스템들의 표본으로 많이 선택되어져 왔다. 예를 들면 2족 보행로봇 이라든지 로켓의 공격 목표 조준과 같은 시스템 같은 것들이 있다. 따라서 도립진자 시스템은 고전 제어 이론이나 현대제어 이론들의 검증을 위해서 많이 사용되어져 왔다.

4.1 시스템 구성

그림 2는 모의실험에 사용한 도립 진자 시스템의 구성을 나타내고 있다.

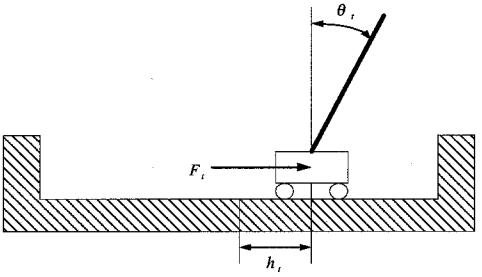

그림 2. 도립진자의 구성

또한 모의실험에서 도립진자의 움직임은 다음의 방정식을 사용했다.

$$\ddot{\theta}_t = \frac{g\sin\theta_t + \cos\theta_t\left[-F_t - m_p l\dot{\theta}_t^2 \sin\theta_t\right]}{l\left[\frac{4}{3} - \frac{m_p \cos^2\theta_t}{m_c + m_p}\right]}$$

$$\ddot{h}_t = \frac{F_t + m_p l\left[\dot{\theta}_t^2 \sin\theta_t - \ddot{\theta}_t \cos\theta_t\right]}{m_c + m_p}$$

여기서,

h_t = 트랙에 대한 차륜의 수평위치 (m)

$\dot{h}_t$ = 차륜의 수평 속도 (m/s)

θ_t = 도립진자가 수직에 대해 기울어진 각도 (degree)

$\dot{\theta}_t$ = 도립진자의 각속도 (deg/s)

F_t = $\pm 10Nt$ = 행동 회로망의 출력

m_c = $1.0kg$ = 차륜의 질량

m_p = $0.1kg$ = 도립진자의 질량

l = $0.5m$ = 도립진자의 질량 중심의 거리

g = $9.8m/s^2$ = 중력가속도

도립진자 시스템의 자세를 유지하기 위해서 시스템에 가해지는 행동은 단지 좌우로의 일정한 크기로 고정된 힘의 시퀀스뿐이고 크기가 0인 힘은 허용되지 않는다. 여기서 시스템의 성능을 평가해주는 신호로 실패신호를 사용하였는데 이 신호는 차륜이 트랙의 경계에 부딪혔을 때나 도립진자가 제어 가능 범위를 벗어났을 경우에 발생하며 다음

과 같이 정의했다.

$$r[t] = \begin{cases} -1, & \text{if } |\theta_t| > 12^\circ \text{ or } |h_t| > 2.4m \\ 0, & otherwise \end{cases}$$

4.2 인공신경회로망의 구성

이 논문에서는 성능평가 신경회로망(evaluation network)과 동작 신경회로망(action network)으로 구성되어진 제어기를 사용하였으며, 각각의 신경회로망은 하나의 은닉층(Hidden Layer)과 하나의 출력층(Output Layer)으로 이루어진 2층의 인공신경회로망으로 구성되어 있다. 성능평가 신경회로망은 도립진자의 상태의 평가 함수를 학습하게 되고, 동작 신경회로망은 시스템의 동작을 생성하게 된다. 학습이 이루어지는 동안에 이 두 인공신경회로망은 동시에 조절된다. 그림 3은 인공신경회로망의 구성이다.

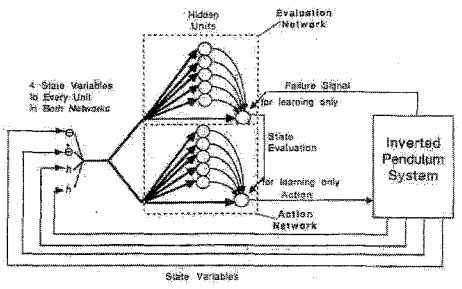

그림 3. 인공신경회로망의 구성

인공신경회로망으로의 입력으로는 도립진자 시스템의 상태를 나타내는 4개의 실수변수와 도립진자시스템의 실패신호(failure signal)이다. 여기서 도립진자 시스템의 동역학과 같은 사전지식은 사용되지 않고 있다. 각각의 상태 입력은 0부터 1사이의 값으로 정규화 시켜서 사용을 하였으며 Charles W. Anderson이 제안한 방법[1]에서는 상태입력 4개의 실수변수 외에 0.5의 값을 갖는 하나의 더미변수를 하나 더 사용하였다. 하지만 이 더미변수의 의미는 설명하지 않고 있으며, 단지 신경회로망의 학습 속도를 빠르게 하고자 하는 의도로 추측해 볼 수 있을 뿐이다. 이 더미변수 하나가 더 추가됨으로 인해서 실제 인공신경회로망을 구현하는데 있어서 수행해야 하는 연산이 늘어나게 되므로 이 논문에서는 이 더미변수를 제거하고 실험을 하였다. 이 더미변수를 제거함으로써 수행해야 하는 연산은 줄어들게 된다.

입력 변수는 다음과 같이 정규화 하였다.

$$x_1[t] = \frac{1}{4.8}(h[t] + 2.4)$$
$$x_2[t] = \frac{1}{3}(\dot{h}[t] + 1.5)$$
$$x_3[t] = \frac{1}{24}(\theta[t] + 12)$$
$$x_4[t] = \frac{1}{230}(\dot{\theta}[t] + 115)$$

이와 같이 입력을 정규화 한 이유는 각각의 입력 변수의 스케일 차이로 인해서 학습에 있어서 상대적으로 큰 값을 갖는 변수에 의해 주도되지 않고 균등하게 학습에 영향을 미치도록 하기 위함이다.

4.3 인공신경회로망의 학습

그림 4는 성능평가 신경회로망의 구성을 나타낸 것이다.

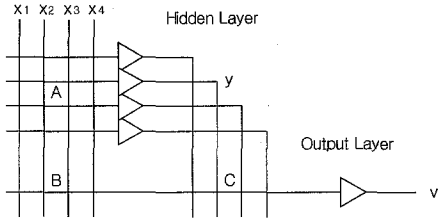

그림 4. 성능평가 신경회로망의 구성

그림에서 A, B, C는 각각 신경회로망의 가중치를 나타내는 행렬을 의미하며 $a_{i,j}$, b_i, c_i의 형태로 각각의 가중치를 나타냈다. 성능평가 신경회로망의 학습은 다음의 방법으로 이루어진다.

출력 : 상태 평가

$$y_i[t_1, t_2] = g\left(\sum_{j=1}^{4} a_{ij}[t_1][t_2]\right),$$
$$v[t_1, t_2] = \sum_{i=1}^{4} b_i[t_1][t_2] + \sum_{i=1}^{4} c_i[t_1] y_i[t_1][t_2]$$
$$g(s) = \frac{1}{1+e^{-s}}$$

동작 평가 : 실패신호와 상태 평가의 변화

$$\hat{r}[t+1] = \begin{cases} 0, & \text{t+1에서 상태가 초기상태} \\ r[t+1] - v[t,t], & \text{t+1에서 상태가 실패상태} \\ r[t+1] + \gamma v[t,t+1] - v[t,t], & otherwise \end{cases}$$

가중치 행렬의 갱신

$$b_i[t+1] = b_i[t] + \beta \hat{r}[t+1] x_i[t],$$
$$c_i[t+1] = c_i[t] + \beta \hat{r}[t+1] y_i[t,t],$$
$$a_{ij}[t+1] = a_{ij}[t] + \beta_h \hat{r}[t+1] y_i[t,t] \cdot (1 - y_i[t,t]) sgn(c_i[t]) x_j[t],$$

파라미터

$0 < \gamma \leq 1; \quad \beta, \beta_h > 0$

그림 5는 동작 신경회로망의 구성을 나타낸 것이다.

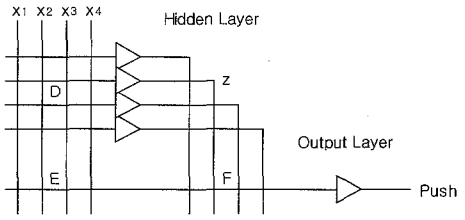

그림 5. 동작 신경회로망의 구성

그림에서 D, E, F는 각각 신경회로망의 가중치를 나타내는 행렬을 의미하며 $d_{i,j}$, e_i, f_i의 형태로 각각의 가중치를 나타냈다. 성능평가 신경회로망의 학습은 다음의 방법으로 이루어진다.

출력 : 동작

$$z_i[t] = g(\sum_{j=1}^{4} d_{ij}[t]x_j[t]),$$

$$p[t] = g(\sum_{i=1}^{4} e_i[t]x_i[t] + \sum_{i=1}^{4} f_i[t]z_i[t]),$$

$$q[t] = \begin{cases} 1, & \textit{with probability } p[t] \\ 0, & \textit{with probability } 1-p[t] \end{cases}$$

$$Push[t] = \begin{cases} 10, & \text{if } q[t]=1; \\ -10, & \text{if } q[t]=0 \end{cases}$$

가중치 행렬의 갱신

$$e_i[t+1] = e_i[t] + \rho\hat{r}[t+1] \cdot (q[t]-p[t])x_i[t],$$

$$f_i[t+1] = f_i[t] + \rho\hat{r}[t+1] \cdot (q[t]-p[t])z_i[t],$$

$$d_{ij}[t+1] = d_{ij}[t] + \rho_h\hat{r}[t+1]z_i[t] \cdot (1-z_i[t])sgn(f_i[t]) \cdot (q[t]-p[t])x_j[t]$$

파라미터

$\rho, \ \rho_h > 0$

4.3 모의 실험 결과

그림 6은 Charles W. Anderson이 제안한 방법[1]의 모의실험 결과를 나타내는 그래프이고 그림 7은 이 논문에서 수행한 모의실험 결과를 나타내는 그래프이다. 두 그림에서 가로축은 신경회로망의 학습횟수이고, 세로축은 실패신호가 발생할 때까지 도립진자의 자세가 유지된 시간단위를 의미한다.

두 그림은 각각 10번의 모의실험 결과의 평균치를 그래프로 나타낸 것인데 두 그림을 비교해 보면, 양쪽 다 5000번 정도의 학습이후에 모의실험에서 설정한 한계시간 이상 도립진자의 자세를 제어하는 것을 볼 수 있다. 따라서 두 방법 모두 도립진자의 자세 제어라는 본래의 목적에 부합하는 결과를 보여주었으며, Charles W. Anderson이 제안한 방법[1]에 있어서 의미의 언급 없이 추가로 입력에 더해졌던 더미입력의 경우 포함하지 않아도 목적에 부합하는 결과가 나온 것을 볼 수 있다.

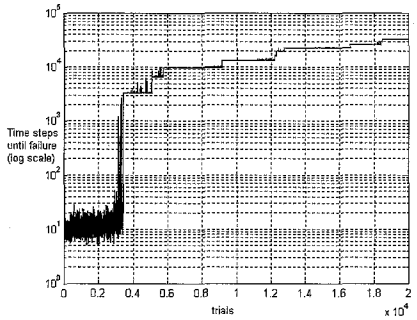

그림 6. 참고 모의실험결과

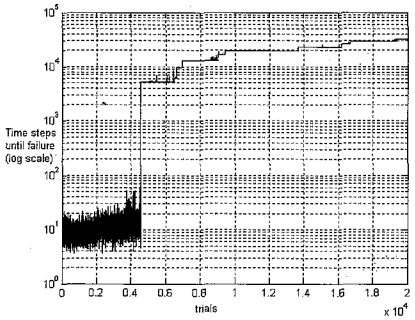

그림 7. 논문의 모의실험결과

5. 결론

이 논문에서는 인공신경회로망의 강화학습을 통한 차륜형 도립진자의 자세 제어를 수행하였다. 기존에 제시되었던 방법에 있어서의 불필요한 요소를 제거시켜서 좀 더 단순화되고 실제 제어기가 구현되어 제어를 수행할 때 필요한 연산의 양을 줄인 방법을 제시하였으며, 모의실험을 통해 기존의 방법의 결과와 이 논문에서 수행한 방법의 결과를 비교해 보았다. 기존의 방법도 물론 도립진자의 자세를 제어한다는 목적은 이룰 수 있었지만 이 논문에서 제시한 방법과 비교하면 불필요한 요소가 포함되어 있었다는 것을 알 수 있었다. 기존의 방법에서나 본 논문에서나 모의 실험을 통한 제어기의 성능 평가가 이루어졌는데, 실제 제어기의 성능의 평가는 실제 실험을 통해서만 가능하다고 보며, 그것이 앞으로 해야할 과제이다.

참 고 문 헌

[1] Charles W. Anderson, "Strategy Learning with Multilayer Connectionist Representations", *proceedings of the Fourth International Workshop on Machin Learning*, pp.103-114, 1987

[2] Richard S. Sutton. "Temporal Credit Assignment in Reinforcement Learning", *PhD thesis*, University of Massachusetts, 1984

[3] 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, SMC-13(5)*, pp.834-846, 1983

[4] Ronald J. Williams and Leemon C. Baird, III, "Analysis of some incremental variants of policy iteration: First steps toward understanding actor-critic learning systems.", *Technical Report NU-CCS-93-11*, Northeastern University, 1993

[5] Richard S. Sutton, "Learning to predict by the method of temporal differences.", *Machine Learning*, 3(1), pp.9-44, 1988

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

[7] Leslie Pack Kaelbling and Andrew W. Moore, *Reinforcement Learning: A Survey*, AI Access Foundation and Morgan Kaufmann Publishers, 1996

[8] Richard S. Sutton and Andrew G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998