퍼지 및 지능시스템학회 논문지 2005, Vol. 15, No. 7, pp. 893-898

RLS 기반 Actor-Critic 학습을 이용한 로봇이동

Robot Locomotion via RLS-based Actor-Critic Learning

김종호, 강대성, 박주영

Jongho Kim, Daesung Kang, and Jooyoung Park

고려대학교 제어계측공학과

요 약

강화학습 방법론 중 하나의 부류인 액터-크리틱 알고리즘은 제어입력 선택 문제에 있어서 최소한의 계산만을 필요로 하고, 확률적 정책을 명시적으로 다룰 수 있는 장점 때문에 최근에 인공지능 분야에서 많은 관심을 끌고 있다. 액터-크리틱 네트워크는 제어입력 선택 전략을 위한 액터 네트워크와 가치 함수 근사를 위한 크리틱 네트워크로 구성되며, 우수한 제어입력의 선택과 정확한 가치 함수 근사를 최대한 신속하게 달성하기 위하여, 학습 과정 동안 액터와 크리틱은 자신들의 파라미터 벡터를 적응적으로 변화시키는 전략을 구사한다. 본 논문은 크리틱의 학습을 위해 빠른 수렴성을 보장하는 RLS (Recursive Least Square)를 사용하고, 액터의 학습을 위해 정책의 기울기(Policy Gradient)를 이용하는 새로운 종류의 알고리즘을 고려한다. 고려된 알고리즘의 적용 가능성은 두개의 링크를 갖는 로봇에 대한 실험을 통하여 예시된다.

Abstract

Due to the merits that only a small amount of computation is needed for solutions and stochastic policies can be handled explicitly, the actor-critic algorithm, which is a class of reinforcement learning methods, has recently attracted a lot of interests in the area of artificial intelligence. The actor-critic network composes of the actor network for selecting control inputs and the critic network for estimating value functions, and in its training stage, the actor and critic networks take the strategy of changing their parameters adaptively in order to select excellent control inputs and yield accurate approximation for value functions as fast as possible. In this paper, we consider a new actor-critic algorithm employing an RLS(Recursive Least Square) method for critic learning, and policy gradients for actor learning. The applicability of the considered algorithm is illustrated with experiments on the two linked robot arm.

Key word : 강화학습(Reinforcement Learning), RLS, 액터-크리틱 학습, 정책의 기울기

1. 서 론

강화학습 방법론 중 하나의 부류인 액터-크리틱 알고리즘은 일반적으로 제어입력 선택 문제에 있어서 최소한의 계산만을 필요로 하고, 확률적 정책을 명시적으로 다룰 수 있는 장점 때문에 최근에 인공지능 및 기계학습 분야에서 많은 관심을 끌고 있다. 액터-크리틱 학습 방법은 정책 반복을 통해 액터와 크리틱의 파라미터를 개선하며, 개선된 파라미터들은 제어 입력을 선택하는데 사용된다. 액터-크리틱 네트워크는, 제어입력 선택 전략을 위한 액터 네트워크와 가치 함수 근사를 위한 크리틱 네트워크로 구성된다. 액터의 학습은 정책의 조정과 관련된 부분으로 현재 상태에서 취할 수 있는 최적의 제어 입력을 선택하는 과정이다. 그리고, 크리틱 학습은 정책의 평가에 관련된 부분으로 현재 상태와 다음 상태의 가치 함수(Value Function) 차이 등을 활용하여 가치 함수를 근사하며, 이 결과로 얻어진 가치 함수에 대한 근사값은 우수한 제어 입력을 선택하는데 이용된다. 우수한 제어입력의 선택과 정확한 가치 함수의 근사를 최대한 신속하게 달성하기 위하여 각 네트워크는 다양한 적응적 전략을 사용하여 각각의 파라미터를 변화시킨다. 본 논문에서는, 액터의 파라미터 갱신을 위하여 정책 기울기(Policy Gradient)를 사용하고 크리틱의 파라미터 갱신을 위하여 RLS(Recursive Least Squares) 기법을 사용하는 알고리즘을 고려한다. 고려된 알고리즘의 적용 가능성은 두개의 링크를 갖는 로봇에 대한 실험을 통하여 예시된다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 정책 기울기 방법을 소개하고 크리틱을 위한 RLS 기법을 이용하여 수정된 액터-크리틱 학습 방법을 제안한다.

3장에서는 제안된 학습 방법을 로봇에 적용했을 경우에 대한 결과를 설명하고 마지막 4장에서는 결과 및 향후 연구 방향을 제시한다.

접수일자 : 2005년 10월 21일
완료일자 : 2005년 12월 5일

2. Policy Gradient와 RLS 기법을 이용한 학습 방법

강화학습은 제어기(Agent)와 주어진 시스템(Environment)의 상호 작용에 따른 보상값과 상태정보를 이용하여 최적의 제어입력을 찾아가는 학습 방법이다. 일반적으로 강화학습은 대상의 상태 전이 확률 $p(x'|x,a) \triangleq \Pr(x_t=x, a_t=a)$과 행동에 따른 보상값 $r(x,a)=E(r_t|x_t=x, a_t=a)$ 을 이용하여 설명할 수 있다. 제어기를 선택하는 데 염두를 두어야 하는 목표는, 정책을 따르면서 할인된 보상값의 합으로 표현되는 목적 함수의 기댓값을 최대화 하는데 있다. 한편 정책 $\pi(a|x)=\Pr(a_t=a|x_t=x)$는 확률밀도 함수로 정의 할 수 있

으며, 가치 함수와 입력 가치 함수(Action Value Function)를 이용한 할인된 상태 분포 $d^{\pi}=\sum_{k=0}^{\infty}\gamma^{k}\Pr(x_t=x|x_0,\pi)$를 갖는 목적함수는 다음과 같이 표현된다.

$$J(\pi)=V^{\pi}(x_0)=\sum_{a}\pi(a|s_0)Q^{\pi}(s_0,a)$$
$$=\sum_{s}d^{\pi}(x)\sum_{a}\pi(a|x)r(x,a) \quad (2.1)$$

2.1 RLS 기법을 이용한 크리틱 학습

참고 논문 [4][5]에서 언급된 것처럼, 최소 자승법(Least Square)은 데이터의 비효율적인 활용과 학습에 사용되는 파라미터 선택 문제를 해결하기 위한 방법이다. 일반적으로 최소 자승법은 다음과 같은 선형 시스템의 해를 찾는 문제로 접근할 수 있다.

$$AW\approx b,\ W\text{는 연결강도 벡터} \quad (2.2)$$
$$(A\in R^{K\times K},\ K\text{는 feature 벡터의 수})$$

참고문헌 [1]과 [3] 등에서 볼 수 있듯이, 강화학습의 적용 중 가치 함수를 근사하는 과정은 다음과 같은 목적함수를 최소화하는 최소자승문제를 해결하는 작업을 필요로 한다.

$$J=\|\sum_{t=1}^{T}A(X_t)W-\sum_{t=1}^{T}b(X_t)\|^2 \quad (2.3)$$
$$A(x_t)\in R^{n\times n},\ b(X_t)\in R^{n},\ \|\cdot\|\text{ 유클리드 놈(norm)}$$

위의 식으로 표현된 목적함수를 최소화하기 위하여, 참고문헌 [4]와 [5]의 LS-TD(Least Square Temporary Difference)를 이용하면, 해의 형태는 다음과 같아진다.

$$W_{LS-TD(\lambda)}=A_T^{-1}b_T=(\sum_{t=1}^{T}A(X_t))^{-1}(\sum_{t=1}^{T}b(X_t)),$$

$$A_T=\sum_{t=0}^{T}(A(X_t))=\sum_{t=0}^{T}z_t(\phi'(x_t)-\gamma\phi'(x_{t+1})) \quad (2.4)$$
$$b_t=\sum_{t=0}^{T}b(X_t)=\sum_{t=0}^{T}z_tr_t$$

예전 상태 정보를 현재 파라미터 개선에 이용하는 적격성을 고려한 식 (2.4)는 아래와 같이 표현할 수 있다.

$$b_t:=b_t+z_tr_t$$
$$A_t:=A_t+z_t(\phi(x_t)-\phi(x_{t+1}))'$$
$$z_{t+1}:=\lambda z_t+\phi(x_t) \quad (2.5)$$

식 (2.5)의 $\lambda\in[0,1]$는 적격성 상수를 z_t는 이전상태를 기억하기 위해 도입된 적격성 트레이스 벡터를 의미하며, ϕ_t는 가치 함수 근사를 위한 기저 벡터를 나타낸다. 식 (2.4)에서 A_t의 값은 충분한 데이터가 모이기 전에 역행렬을 구할 수 없는 경우가 종종 발생한다. 이를 극복하기 위해 A_0의 값은 다음과 같이 나타낼 수 있다.

$$A_0=\delta I+\phi(x_t)(\gamma\phi(x_{t+1})-\phi(x_t)) \quad (2.6)$$

식 (2.3)으로 표현되는 에러의 값을 최소하기 위해 각 스텝마다 파라미터 벡터 W를 개선하게 되는데, 이 과정에서 다음의 Bellman 방정식 [7]이 이용하게 된다.

$$Q^{\pi}(x,\mu)=r(x,\mu)+\gamma\int_{X}p(x'|x,\mu)V^{\pi}(x')dx \quad (2.7)$$

식 (2.7)의 $r(x,\mu)$는 즉각적인 보상값(Immediate rewards)을 나타내며, $p(x'|x,\mu)$는 상태 전이 확률을 의미한다. 참고 논문[4]에서 가치 함수와 입력 가치 함수의 차를 어드밴티지 가치 함수(Advantage Value Function)라 정의하고 다음과 같이 근사하였다.

$$A^{\pi}(x,\mu)=Q^{\pi}(x,\mu)-V^{\pi}(x) \quad (2.8)$$
$$\approx\nabla_{\theta}\log\pi(a|x_t)w$$

위의 식 (2.8)에 언급된 바와 같이, 어드밴티지 가치 함수는 양립 근사자(Compatible Function Approximator)로 근사되며, 이 근사 함수는 파라미터 벡터 w에 선형임을 알 수 있다.

크리틱에서 다루어지는 목적함수는, 식 (2.7)과 식 (2.8)을 이용하여 다음과 같은 최소 자승문제로 표현할 수 있다.

$$\Psi_t(v,w)$$
$$\doteq\|\sum_{k=0}^{t}z_k[(V_v(x_k)+A_w(x_k,a_k))-(r_k+\gamma V(x_{k+1})]\|^2$$
$$=\|\sum_{k=0}^{t}z_k[\phi'(x_k)-\gamma\phi'(x_{k+1}),\nabla_{\theta}\log\pi(a_k|x_k')]\begin{bmatrix}v\\w\end{bmatrix}-$$
$$\sum_{k=0}^{t}z_kr_k\|^2 \quad (2.9)$$

식 (2.9)에서 나타나는 z_k는 예전 상태를 기억하기 위한 적격성 벡터로, 다음과 같은 형태로 정의된다.

$$z_k=\gamma\lambda z_{k-1}+[\phi'(x_t),\nabla_{\theta}\log\pi(a_k|x_k)']'\quad\text{for } k\geq1$$
$$z_0=[\phi'(x_0),\nabla_{\theta}\log\pi(a_0|x_0)']' \quad (2.10)$$

한편, 매트릭스 A_t의 역행렬을 구하는 과정에, 참고문헌 [3]과 [8] 등에 언급된 다음의 매트릭스 역행렬 공식을 활용하여 식 (2.9)에 대한 RLS형태의 해를 구할 수 있다.

$$(A+BC)^{-1}=A^{-1}-A^{-1}B(I+CA^{-1}B)^{-1}CA^{-1} \quad (2.11)$$

식 (2.11)의 매트릭스 역 공식과 식 (2.9)을 결합한 형태는 다음과 같다.

$$z_0=[\phi'(x_0),\nabla_{\theta}\log\pi(a_0|x_0)']'$$
$$A_0=\delta I+z_0[\phi'(x_0)-\gamma\phi'(x_1),\nabla_{\theta}\log\pi(a_0|x_0)']$$
$$A_t=\beta A_{t-1}+z_t[\phi'(x_t),\nabla_{\theta}\log\pi(a_t|x_t)]$$
$$P_t=A_t^{-1}\quad\text{for } t\geq0\quad K_t=P_tz_t\quad\text{for } t\geq0 \quad (2.12)$$

$$z_t=\gamma\lambda z_{t-1}+[\phi'(x_t),\nabla_{\theta}\log\pi(a_t|x_t)]$$
$$P_t=\frac{1}{\beta}(P_{t-1}-\frac{P_{t-1}z_t[\phi'(x_t),\nabla_{\theta}\log\pi(a_t|x_t)]z_t}{\beta+[\phi'(x_t),\nabla_{\theta}\log\pi(a_t|x_t)]P_{t-1}z_t})$$
$$K_t=\frac{P_{t-1}z_t}{\beta+[\phi'(x_t),\nabla_{\theta}\log\pi(a_t|x_t)]'P_{t-1}z_t}$$
$$\begin{bmatrix}v_t\\w_t\end{bmatrix}=\begin{bmatrix}v_{t-1}\\w_{t-1}\end{bmatrix}+K_t(r_t-$$
$$[\phi'(x_t)-\gamma\phi'(x_{t+1}),\nabla_{\theta}\log\pi(a_t|x_t)']\begin{bmatrix}v_{t-1}\\w_{t-1}\end{bmatrix}]$$

식 (2.12)에 나타나는 $\beta\in[0,1]$의 값은 이전 상태 정보를 어떻게 활용할 것인지를 나타내는 상수에 해당한다. $\beta=0$인

경우 A_t에 기억되어 있는 값들을 전혀 고려하지 않고, 현재 정보만을 이용해 이를 학습에 반영하는 경우에 경우를 의미하며, $\beta=1$ 경우에는 이전 상태의 모든 정보를 기억하고 이를 학습에 이용하는 경우에 해당한다.

2.2 정책 기울기를 이용한 액터 학습

액터-크리틱 학습에서 액터는 보상값의 합으로 표현되는 목적함수를 최대화 하는 방향으로 액터의 파라미터 벡터를 개선하게 된다.

제어 입력을 선택하는 조건부 확률 $\pi_\theta(a|x)$에서 θ는 분산을 결정하는 특징 벡터에 나타낸다. 이에 따라 액터가 목적으로 하는 함수 형태는 다음과 같이 정의 할 수 있다.

$$J(\pi)=J(\theta)=\sum_x d^{\pi_\theta}(x)\sum_a \pi_\theta(a|x)r(s,a) \tag{2.13}$$

함수 근사를 위한 일반적인 학습은 식 (2.13)의 목적함수의 기울기를 따르는 방법이다. 참고 논문 [2][7]에 언급된 것처럼 정책의 기울기는 다음과 같이 표현된다.

$$\begin{aligned}\nabla_\theta J(\theta)&=\sum_s d^{\pi_\theta}(x)\sum_a \nabla_\theta \pi_\theta(a|x)Q^{\pi_\theta}(x,a)\\&=\sum_s d^{\pi_\theta}(x)\sum_a \nabla_\theta \pi_\theta(a|x)(Q^{\pi_\theta}(x,a)-V^{\pi_\theta}(x))\\&=\sum_s d^{\pi}(x)\sum_a \pi_\theta(a|x)\nabla_\theta \log\pi_\theta(a|s)A^{\pi_\theta}(x,a)\end{aligned} \tag{2.14}$$

식 (2.8)과 같은 양립 근사자를 이용하면 식 (2.14)의 표현은 다음과 같이 나타낼 수 있다.

$$\begin{aligned}\nabla_\theta J(\theta)&=\sum_x d^{\pi_\theta}(x)\sum_a \pi_\theta(a|x)\nabla_\theta \log\pi_\theta(a|x)A^{\pi_\theta}(x,a)\\&\approx\sum_s d^{\pi_\theta}(x)\sum_a \pi_\theta(a|x)\nabla_\theta \log\pi_\theta(a|x)\mathcal{A}_w(x,a)\\&=F(\theta)w.\end{aligned} \tag{2.15}$$

단, $F(\theta)\triangleq\sum_s d^{\pi_\theta}(x)\sum_a \pi_\theta(a|x)\nabla_\theta \log\pi_\theta(a|x)\nabla_\theta \log\pi_\theta(a|x)'$

본 논문에서는, 목적함수 $J(\pi)$를 최적화하기 위하여 목적함수의 기울기(Gradient)를 따라 파라미터 θ를 변화시키는 기울기 상승법을 활용하고, 정책의 기울기를 근사하기 위하여 위의 식 (2.15)를 사용하는 전략을 고려한다. 따라서, 식 (2.15)의 정책의 기울기를 이용한 학습은 다음과 같은 단순한 형태로 표현될 수 있다.

$$\theta\leftarrow\theta+\alpha\nabla_\theta J\approx\theta+\alpha F(\theta)w \tag{2.16a}$$

이 식에서 α는 학습률을 나타내고, θ는 액터의 파라미터 벡터를 의미하며, w는 크리틱 네트워크에서 추정하는 양립 근사자의 파라미터 벡터이다. 액터의 파라미터 θ를 갱신하는 과정에서, 정책의 기울기를 직접 사용하는 방안 대신 고려할 수 있는 또 다른 방법은 natural gradient $\widetilde{\nabla}_\theta J$를 이용하는 학습 방법이다[2]. 이 경우에 참고문헌 [2]에서 밝혀진 $F(\theta)$와 피셔 정보 행렬(Fisher information matrix)이 같아진다는 괄목할만한 사실을 활용하면 θ의 갱신은 다음과 같은 단순한 형태로 표현될 수 있다.

$$\theta\leftarrow\theta+\alpha\widetilde{\nabla}_\theta J\approx\theta+\alpha w \tag{2.16b}$$

본 논문에서는 액터의 파라미터 갱신을 위하여 (2.16a)와 (2.16b)를 사용하는 경우 각각을 모두 고려한다.

2.3 RLS 기반 액터-크리틱 학습

앞 절에서 언급한 크리틱을 위한 RLS와 액터를 위한 정책 기울기 방법을 결합한 알고리즘의 형태는 다음과 같다.

(1) 관련된 모든 파라미터를 초기화함
(2) 시간 스텝 t의 관측 변수 x_t관찰
(3) 확률분포 $\pi(a|x)=\Pr(a_t=a|x_t=x)$에 따라 제어 입력 μ_t를 샘플링하여 실행
(4) 제어입력에 따른 다음 상태변수(x_{t+1})와 보상값(r_t) 관찰
(5) 파라미터 개선
$\zeta_t\leftarrow[\phi'(x_t)-\gamma\phi(x_{t+1}),\nabla_{w_\theta}\log\pi(a_t|x_t)']$를 이용하여 다음의 업데이트를 수행함.
(5-a) 크리틱 파라미터 개선

$$z_t=\gamma\lambda z_{t-1}+\zeta_t'$$

$$P_t=\frac{1}{\beta}\left(P_{t-1}-\frac{P_{t-1}z_t\zeta_t P_{t-1}}{\beta+\zeta_t P_{t-1}z_t}\right)$$

$$K_t=\frac{P_{t-1}z_t}{\beta+\zeta_t P_{t-1}z_t}$$

$$\begin{bmatrix}v_{\theta_t}\\w_{\theta_t}\end{bmatrix}=\begin{bmatrix}v_{\theta_{t-1}}\\w_{\theta_{t-1}}\end{bmatrix}+K_t\left(r_t-\zeta_t\begin{bmatrix}v_{\theta_{t-1}}\\w_{\theta_{t-1}}\end{bmatrix}\right)$$

(5-b) 액터 파라미터 개선
$\theta_{t+1}\leftarrow\theta_t+\alpha F(\theta_t)\theta_t$ 또는 $\theta_{t+1}\leftarrow\theta_t+\alpha\theta_t$
(6) Go to step (2)

3. 모의 실험

3.1 Kimura 로봇

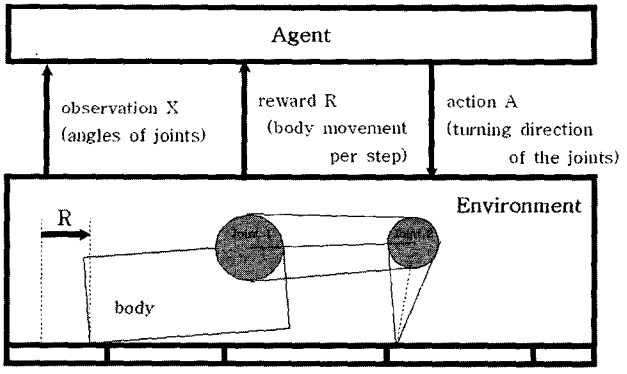

그림 1. Kimura의 기는 로봇[6]

참고문헌 [7]에서 Kimura 등은 강화학습의 효용성을 보이기 위해 간단한 기는 로봇을 응용 문제로 고려하였다. 이 로봇은, 중력이 가해지는 환경 아래에서 두 개의 링크를 가지고 기는 동작을 수행하는 평면형 머니퓰레이터(planar manipulator)로써 그림 1의 구조를 갖는다.

이 로봇에 부과된 임무는 최대한 빨리 전진하는 것인데, 제어기(agent)는 로봇 및 환경에 대한 구체적인 모델 또는 정보가 주어지지 않은 상태에서 직접적인 경험을 통해 관찰된 보상값(rewards) r 만을 가지고 효과적인 제어 규칙을 발견해내야 한다. 각 시간 스텝 때마다 에이전트는 조인트의 각도를 읽어 들이고 확률적 제어입력 선택 전략에 따라 조인

트에 연결된 모터의 회전 방향 및 회전각도를 결정한다.

그리고, 학습 과정에서 이용되는 보상값 r 을 위해서는 해당 시간 스텝 동안 전진한 거리가 사용된다. 만일 로봇이 후진하는 경우에는 후진한 거리만큼의 음의 보상값(negative reward)이 생성됨은 물론이다. 직관적으로 생각할 때에, 위의 로봇이 최대한 빨리 전진하기 위해서는 기면서 앞으로 나아가는 패턴을 신속하게 습득해야 함을 알 수 있다. 본 논문에서 고려하는 로봇 관련 데이터는 [7]의 경우와 같다.

본 논문에서 고려하는 로봇 관련 데이터는 [7]의 경우와 같다. 따라서, 로봇의 위쪽 팔의 길이는 34 cm이고(이하, 단위 생략), 아래쪽 팔의 길이는 20이다. 그리고, 몸체와 위쪽 팔을 잇는 첫 번째 조인트는 몸체의 좌측하단 코너로부터 수평방향으로 32, 수직방향으로 18 떨어진 곳에 위치한다. 몸체와 위쪽 팔을 잇는 조인트의 움직임은 몸체와 수평인 방향에서 [-4, 35] 도 범위에서만 가능하고, 위쪽 팔과 아래쪽 팔을 잇는 두 번째 조인트의 움직임은 위쪽 팔과 수평인 방향에서 [-120, 10] 도 범위에서만 가능하다. 그리고, 아래쪽 팔의 뾰족한 끝부분이 지면에 닿아 있을 때에는, 뾰족한 끝부분은 미끄러지지 않고 몸체만 미끄러짐을 가정한다.

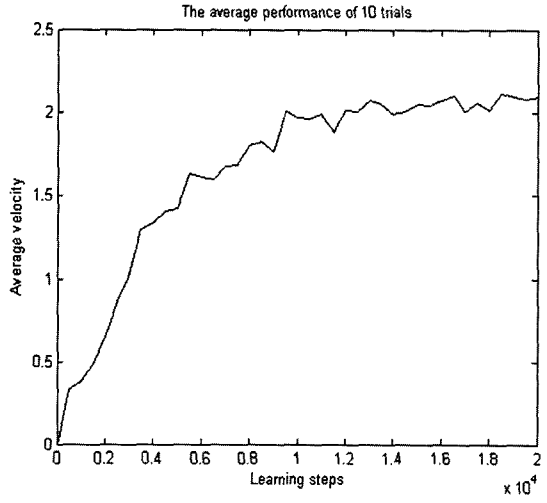

그림 2. Kimura의 로봇을 RPO(λ)-RLS를 적용하여 학습시킨 결과[9]

3.2 학습을 이용한 Kimura 로봇 이동

본 논문에서는 [7]에서의 이론 전개를 참고하여, σ에는 1의 값을 actor에는 각 조인트의 제어입력 선택 전략을 위한 확률분포 φ로 다음과 같은 정규분포를 고려하였다:

$$\varphi(\mu; c) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\mu - c)^2}{2\sigma^2}\right)$$

그리고 각 조인트에서는 로봇의 과도한 움직임을 막기 위해서 각 시간 스텝 당 [-12도 , 12도]범위까지의 움직임만 허용하는 한계성을 부여하였다. 한편 기는 로봇이 받아들이는 입력값은 변위가 [-1,1] 범위가 되도록, 관련 측 변수인 조인트를 적절하게 스케일링한 값을 사용하였다.

본 논문의 가치 함수 근사 과정에서 사용된 기저 벡터는 다음과 같다. 각 조인트의 스케일링 값인 θ_1과 θ_2를 입력으로 하고 고정된 분산과 평균을 갖는 RBF(Radial Basis Function)을 사용하여 각 조인트가 갖는 RBF함수의 값으로 나눈 NRBF(Normalized Radial Basis Function)를 기저 벡터로 사용하였다.

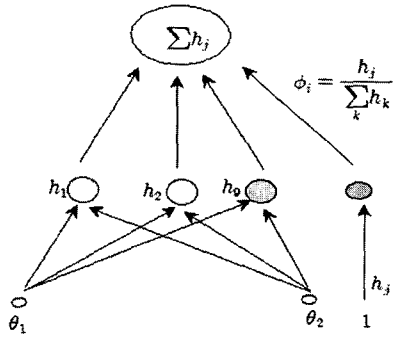

그림 3. NRBF Network를 이용한 가치함수 근사

실험에서는 10번의 episode를 실행했으며, 각 episode는 모두 20000번의 step으로 구성되어 있다. 평균속도는 각 500step의 배수에 그동안 학습된 actor의 파라미터를 이용하여, 한정된 거리를 이동하게 한 후 그에 대한 평균속도를 구했다.

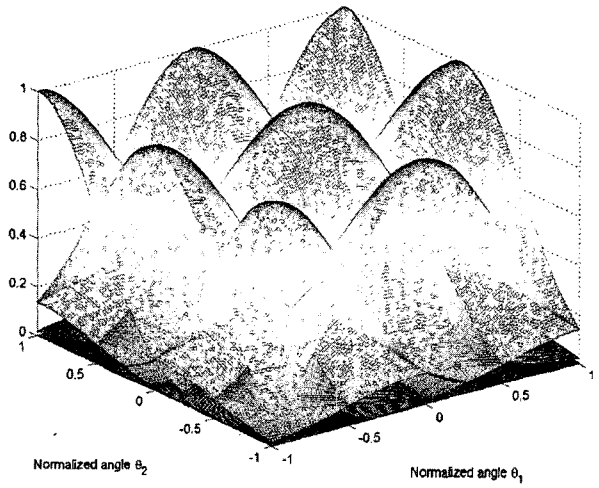

그림 4. NRBF Network에 사용된 기저 함수

학습에서 사용된 그 밖의 관련 파라미터는 다음과 같다.

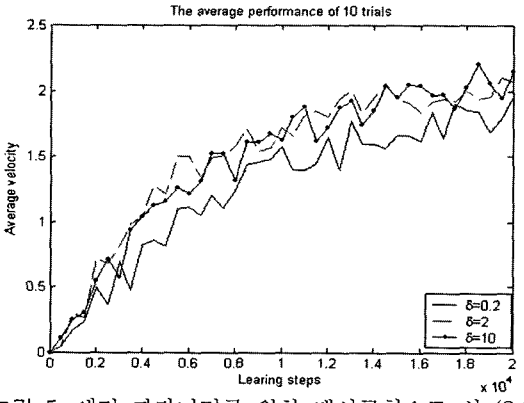

그림 5. 액터 파라미터를 위한 갱신규칙으로 식 (2.16a)를 사용한 경우의 평균속도

- 할인율 $\gamma = 0.95$
- 감쇠율 $\lambda = 0.75$
- 학습율 $\alpha = 0.003$

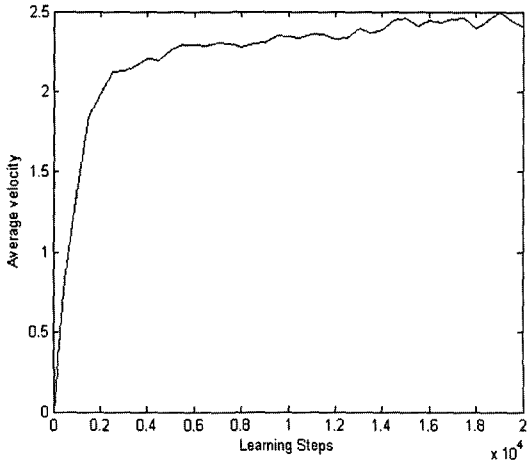

그림 6. 그림 5. 액터 파라미터를 위한 갱신규칙으로 식 (2.16b)를 사용한 경우의 평균속도

그림 5에는, 액터 파라미터 갱신에 (2.16a)를 사용한 경우에 대해 δ 값을 바꾸어가면서 얻은 결과를 정리하였다. 그림에서 볼 수 있듯이 δ의 변화는 로봇의 평균 이동속도에 큰 영향을 주지는 않는 것으로 밝혀졌으며, 참고문헌 [6]에 소개된 경우보다 우수한 성능이 관찰되고 최근에 발표된 저자의 논문 [9]의 경우와 유사한 성능이 얻어짐을 확인할 수 있었다. 그림 6에는 δ 값을 위하여 0.5를 사용하고, 액터 파라미터를 위한 갱신규칙으로 식 (2.16b)를 사용한 경우의 평균속도를 그렸다. 그림 2, 5와 6을 비교함으로부터 관찰할 수 있듯이, 액터를 위한 학습으로 natural gradient 방법을 이용한 학습 방법이 기존의 방법론뿐만 아니라 단순한 policy gradient를 이용한 학습 방법보다 평균 속도 면에서 우수한 성능을 제공함을 확인할 수 있다.

4. 결론 및 향후 과제

액터-크리틱 방법을 대상으로 하여, 액터를 위한 학습으로 natural gradient 또는 gradient를 사용하고 크리틱을 위한 학습으로 RLS(Recursive Least Square)를 이용한 학습 방법이 기존의 학습 방법보다 우수함을 실험을 통하여 관찰하였다. 이는, 빠른 연산으로 최소자승문제의 해를 찾는 RLS 기법이 강화학습에 효과적으로 적용될 수 있는 주요한 사례를 제공하는 의미를 가진다.

향후 과제로는 제안한 학습 방법을 여러 종류의 응용문제에 광범위하게 적용해 보는 문제와, 최근 기계학습 분야에 큰 영향을 미치고 있는 커널 기법을 본 논문에서 고려한 방법론에 접목 시켜보는 문제 등을 들 수 있다.

참 고 문 헌

[1] A. Nedic and D. P. Bertsekas "Least square policy evaluation algorithms with linear function approximation", Journal of Discrete Event Dynamic Systems, Vol. 13, pp. 79-110, 2003.

[2] J. Peters, S. Vijayakumar and S. Schaal "Reinforcement learning for humanoid robotics," Proceedings of 3rd IEEE-RAS International Conference on Humanoid Robots, Karlsruhe, Germany, 2003.

[3] X. Xu, H. He and D. Hu, "Efficient reinforcement learning using recursive least-Square methods," Journal of Artificial Intelligence Research, vol 16, pp. 259-292, 2002

[4] J. Boyan, "Least-squares temporal difference learning." Proceedings of the sixteenth International Conference(ICML), pp. 49-56, 1999.

[5] J. Boyan, "Technical update: least-squares temporal difference learning", Machine Learning, vol. 49, pp. 233-246, 2002.

[6] H. Kimura, K. Miyazaki, and S. Kobayashi, "Reinforcement learing in POMDPs with function approximation," Proceedings of the 14th International Conference on Machine Learning (ICML '97), pp. 152-160, 1997.

[7] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, MIT Press, 1998.

[8] L. Ljung, "Analysis of recursive stochastic algorithm," IEEE Transactions on Automatic Control, vol, 22, pp. 551-575, 1977.

[9] 김종호, 강대성, 박주영, "RPO기반 강화학습 알고리즘을 이용한 로봇제어" 한국 퍼지 및 지능 시템 학회 2005년도 춘계학술 대회 논문집, 15권 1호, pp, 505-507, 2005년 4월.

저 자 소 개

김종호(Jongho Kim)
2004년 : 고려대학교 제어계측공학과 졸업 (학사)
2004년~현재 : 고려대학교 제어계측공학과 대학원

관심분야 : 강화학습, SVM응용
Phone : 019-601-3420
E-mail : oyeasw@korea.ac.kr

강대성(Daesung Kang)
2004년 : 고려대학교 제어계측공학과 졸업(학사)
2005년~현재 : 고려대학교 제어계측공학과 대학원

관심분야 : SVM, 강화학습
phone : 019-506-2086
E-mail : mpkds@korea.ac.kr

박주영(Jooyoung Park)

1983년 : 서울대학교 전기공학과 졸업(학사)
1985년 : 한국과학기술원 졸업(석사)
1985년 3월~1988년 7월 : 한국전력 월성 원자력발전소 근무
1992년 : University of Texas at Austin 전기 및 컴퓨터공학과 졸업(박사)
1992년 8월~1993년2월 : 한국전력 전력경제연구실 선임전문원
1993년 3월~현재 : 고려대학교 과학기술대학 제어계측 공학과 교수
관심분야 : 신경망이론, 지능시스템, 비선형시스템
E-mail : parkj@korea.ac.kr