Domain Knowledge를 이용한 강화 학습

Reinforcement Learning Algorithm Using Domain Knowledge

。장 시 영*, 공 성 학*, 서 일 홍*, 오 상 록**

*한양대학교 전자공학과(Tel : 82-31-408-5802; Fax : 82-31-408-5803 ;
E-mail: ihsuh@hanyang.ac.kr)
**한국과학기술연구원 지능제어센터(Tel : 82-2-958-5757; Fax : 82-2-958-5749 ;
E-mail: sroh@amadeus.kist.re.kr)

Abstract: Q-Learning is a most widely used reinforcement learning, which addresses the question of how an autonomous agent can learn to choose optimal actions to achieve its goal about any one problem. Q-Learning can acquire optimal control strategies from delayed rewards, even when the agent has no prior knowledge of the effects of its action in the environment. If agent has an ability using previous knowledge, then it is expected that the agent can speed up learning by interacting with environment. We present a novel reinforcement learning method using domain knowledge, which is represented by problem-independent features and their classifiers. Here neural network are implied as knowledge classifiers.

To show that an agent using domain knowledge can have better performance than the agent with standard Q-Learner. Computer simulations are illustrated where a mobile robot navigation problem under complex obstacles is considered.

Keywords: domain knowledge, neighboring state, Q-Learning, neural network, classifier, mobile robot

1. 서론

강화 학습이란 에이전트(Agent)가 알려지지 않은 환경에서 행동과 보답을 주고 받으며, 임의의 상태에서 가장 적합한 행위를 학습하는 방법이다.

Q-Learning은 가장 널리 사용되는 강화 학습 방법들 중에 하나로 이 학습법은 현재 상태에서의 행위를 미래 행위들로부터 얻게 되는 총 보답을 예측하는 행위값에 대응시키는 행위 함수를 학습하는 방법이다. 그러나, Q-Learning은 빠른 실시간 성능을 가질 지라도, 과거에 학습했던 지식을 이용할 수 없다는 단점이 있다. 에이전트가 같은 환경 내에서 여러 문제를 해결해야 할 경우, 과거에 문제들을 해결하면서 얻은 환경에 대한 지식을 활용할 수 있다면 학습시간을 줄일 수 있다. 이러한 환경에 대한 모델을 학습하는 방법에 관한 연구가 진행되어 왔다. 예를 들면, Dyna_Q, Policy reuse, Region-based Q-Learning, Local state feature to bias exploration 등이 있다[5],[4],[6],[1].

이런 여러 방법들 중 Local state feature to bias exploration 방법은 그림1에서와 같이 과거에 학습한 여러 문제들에 대한 환경 지식(Domain Knowledge)을 Local state feature라는 기억공간에 학습시킨 후 에이전트가 행위 함수를 학습할 때 이 지식을 이용할 수 있도록 하는 방법이다[1]. 그러나 이 방법은 에이전트가 적절한 행위 함수를 학습하기 위해서는 local state feature내에 들어 있는 지식의 신뢰도가 높아야 한다는 가정이 필요하다. 본 논문에서는 Local state feature to bias exploration 방법을 자율 이동 로봇에 적용하는 컴퓨터 모의 실험을 하여 에이전트가 좀더 안정된 방법으로 Local state feature를 사용할 수 있도록 개선하고, 개선된 방법의 수렴성 증명과 그 유용성을 모의 실험 결과를 통하여 검증하여 본다.

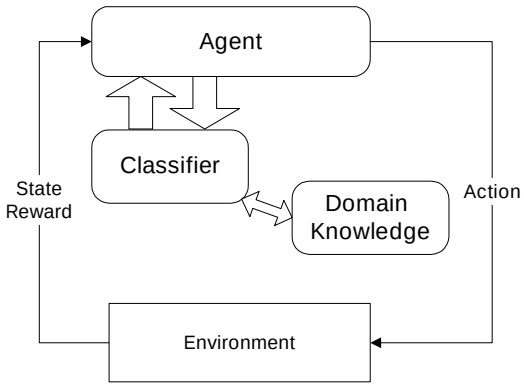

그림1. 환경 지식을 이용하는 강화 학습.
Figure1. Reinforcement Learning Algorithm Using Domain Knowledge.

2. Q-Learning

Q-Learning은 대표적인 off-policy 강화 학습으로 행위 함수 $Q(s_t,a_t)$ 와 행위 책략 $\pi^*(s_t)$ 은 다음과 같다 [2],[3].

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha\left(r_{t+1} + \gamma \max_a \{Q(s_{t+1},a)\} - Q(s_t,a_t)\right) \quad (1)$$

$$\pi^*(s_t) = \arg\max_a Q(s_t,a) \quad (2)$$

3. 환경 지식을 이용한 Q-Learning

환경 지식을 이용한 Q-Learning을 적용하기 위해서는 두 가지 가정이 필요하다. 첫 번째 가정은 환경 내에 정의된 모든 상태들이 Local state feature라는 저장 공간을 가지고 있어야 하고, 두 번째로 에이전트는 충분히 많은 문제를 학습한 상태여야 한다는 것이다. 여기서 문제란 에이전트의 시작위치와 도착위치, 그리고 장애물의 위치들을 포함하는 에이전트가 해결해야 할 수행 과제이다.

3.1. Local state feature

Local state feature는 환경 지식을 추출하기 위한 단위이다. 과거에 학습한 행위 함수를 새로운 문제에 그대로 적용할 수 없는 이유는 그 행위 함수가 과거의 문제에 국한되어 학습이 되어 있기 때문이다. 에이전트가 주어진 문제를 과거에 학습한 환경 지식을 이용하여 해결하기 위해서는 그 환경 지식이 문제와는 독립적인 특성을 가져야 한다. 예를 들면, 에이전트가 선택할 수 있는 행위가 네 방향일 경우에, 현재 에이전트의 주위의 세 방향에 장애물이 있다면 에이전트는 문제와는 관계없이 나머지 한 방향으로만 움직일 수 있다. 또한 에이전트의 좌측에 벽이 있다면 에이전트는 좌측으로 움직이는 행위를 할 수 없다.

Local state feature는 문제와 독립적이고, 현재 에이전트의 상태와 주변 상태들을 관찰함으로써 생성된다.

3.2. Example Set

에이전트가 주어진 문제에 대하여 행위 함수를 학습한 후 환경 지식을 추출하기 위해서는 모든 상태에서의 Local state feature와 각 상태에서 취한 행위, 그리고 그 행위에 대한 평가가 이루어져야 한다. 이 세 가지의 요소를 가진 저장공간을 Example이라 한다. 또한 모든 상태에 대한 Example의 집합을 Example Set이라 한다.

Example의 생성시에 행위에 대한 평가는 에이전트가 각 상태에서 취한 행위에 대한 행위 함수의 값이 그 상태에서 취할 수 있는 모든 행위들과 비교하여 제일 크다면 Positive Example 이라고 하고, 각 상태에서 취한 행위에 대한 행위 함수의 값이 변화가 없다면 Negative Example이라고 한다.

3.3. Classifier

최종적으로 에이전트는 여러 문제들을 통하여 환경 지식을 추출할 때 각 문제를 해결하면서 생성한 Example Set을 이용한다. Classifier는 Example Set의 정보를 가공, 처리하는 역할을 하며, 개수는 에이전트가 취할 수 있는 행위의 개수만큼 있어야 한다. 에이전트는 Example Set내에 하나의 Example을 학습하기 위해서 Example의 행위요소를 참조하고, 행위요소에 해당하는 Classifier에 그 Example을 학습시키게 된다. 여기서 Classifier는 Neural Network을 사용하였으며 Example을 학습시키는 방법은 Back Propagation Learning Method를 택하였다.

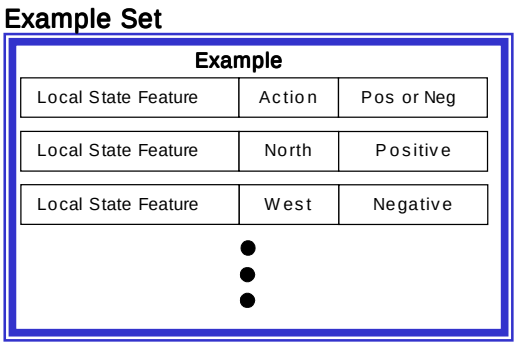

그림2. Example Set의 구성.
Figure2. The Structure of Example Set.

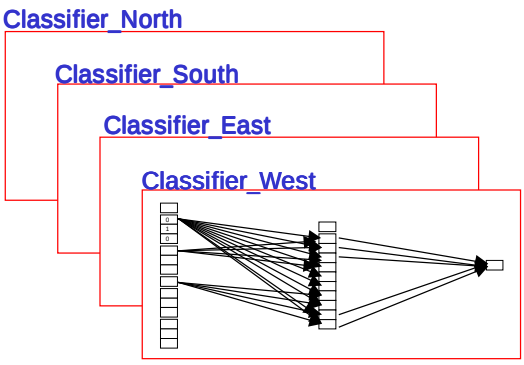

그림3. 행위에 따른 Classifier의 구성.
Figure3. The Structure of Classifiers.

3.4. Exploration bias

에이전트가 새로운 문제에 대하여 행위 함수를 학습할 때 환경 지식을 얻기 위해서는 Classifier를 이용해야 한다.

Q-Learning의 행위 책략 식(2)를 normalization 하면 다음과 같은 식이 된다.

$$\pi^*(s_t) = \arg\max_a P_t(s_t, a) \quad (3)$$

$$P_t(s_t, a) = \frac{Q(s_t, a)}{\sum_a Q(s_t, a)} \quad (4)$$

임의의 상태 s_t 에서 행위 a 에 대한 Classifier의 값 $C_a(s_t)$을 normalization을 하면 다음과 같다.

$$w(s_t, a) = \frac{C_a(s_t)}{\sum_{a'} C_{a'}(s_t)} \quad (5)$$

따라서 새로운 행위 책략은 다음과 같은 식으로 표현할 수 있다.

$$\pi^*(s_t) = \arg\max_a \frac{w(s_t, a) \cdot P_t(s_t, a)}{\sum_{a'} w(s_t, a') \cdot P_t(s_t, a')} \quad (6)$$

4. 개선된 환경 지식을 이용한 Q-Learning

Local state feature to bias exploration 방법이 수렴하기 위해서는 Classifier내에 저장되어 있는 환경 지식의 신뢰도가 높아야 한다는 제약이 있다. 만약 Classifier내에 올바른 지식이 들어 있지 않다면 행위책략에서 선택된 행위는 최적의 행위가 아닐 것이다.

또한, 에이전트가 최적의 행위를 학습하는 과정에서 Classifier의 사용으로 학습 초기에는 속도의 향상을 보인 반면에 학습에 최적에 가까워지는 시기에는 오히려 Q-Learning보다 부진한 성능을 보였다.

Classifier의 주된 역할은 행위 함수들이 학습되어 있지 않은 학습 초기에 적절한 행위를 할 수 있도록 에이전트를 유도해 주는데 있으며, 학습이 최적에 가까워질 수록 빠른 실시간 성능을 가지는 Q-Learning의 장점을 살릴 수 있도록 Classifier의 영향력을 줄이는 개선된 알고리즘을 제안한다.

Q-Learning에 의한 행위의 선택확률과 Classifier에 의한 행위의 선택확률을 동등한 비율로 하는 것이 아니라 Classifier에 의한 행위의 선택확률에 무게(weight)를 두어서 적용하기 위하여 식(5)과 식(6)을 다음과 같이 변형하였다.

$$w_{new}(s_t, a) \leftarrow w(s_t, a) + Value_{weight}(episode) \quad (7)$$

$$\pi^*(s_t) = \arg\max_a \frac{w_{new}(s_t, a) \cdot P_t(s_t, a)}{\sum_{a'} w_{new}(s_t, a') \cdot P_t(s_t, a')} \quad (8)$$

식(7)에서 $Value_{weight}(episode)$ 은 episode에 대한 함수이며, episode가 증가함에 지수 형태로 증가하는 함수이다.

식(7)을 적용한다면 환경 지식에 대한 신뢰도가 낮은 경우에도 에이전트는 안전하게 최적의 행위를 학습할 수 있다. 또한, 에이전트의 학습에 Classifier의 영향이 학습 초기에만 크게 작용하고, 행위 함수들이 최적에 가까워 질수록 적게 작용될 것이다.

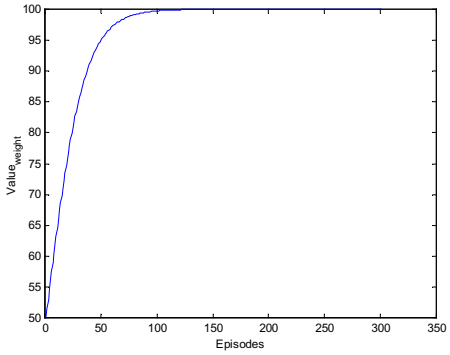

그림4. episode에 따른 $Value_{weight}(episode)$.

Figure4. Function $Value_{weight}$ of episode.

증명 :

$w(s,a)$는 normalization된 값이므로 1보다 작은 값이다. 따라서 episode가 증가함에 따라 $w_{new}(s,a)$는 $Value_{weight}(episode)$의 값에 수렴할 것이다.

$$w_{new}(s_t, a) \approx Value_{weight}(episode) \quad (9)$$

결국, 식(8)은 식(3)과 같은 Watkins에 의해 제안된 Q-Learning의 행위 책략이 됨을 알 수 있다[2],[3].

5. 모의 실험

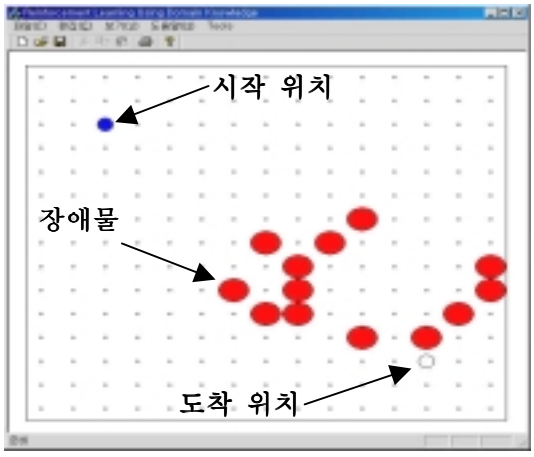

그림5. 15 × 15 map 문제.

Figure5. a problem of 15 × 15 map.

그림 5와 같은 유형의 문제 총 50개(표 1)를 이용하여 에이전트의 환경 지식을 미리 학습시켰다.

표 1. Map의 크기와 문제의 개수.
Table 1. Size of map and number of problems.

Size	7 × 7	10 × 10	12 × 12	15 × 15
개수	20	10	10	10

에이전트의 가능한 행위는 North, South, East, West의 네 방향으로 정하였고, Local state feature는 임의의 상태에서 거리가 4인 상태들을 관찰하여 구성하였으며, 개수는 121개 이다. Classifier에 사용한 Neural Network의 구조는 하나의 숨은 계층(hidden layer)을 가진 구조로 각 입력 계층, 숨은 계층, 그리고, 출력 계층내의 뉴런의 개수는 121개, 20개, 1개로 구성하였다.

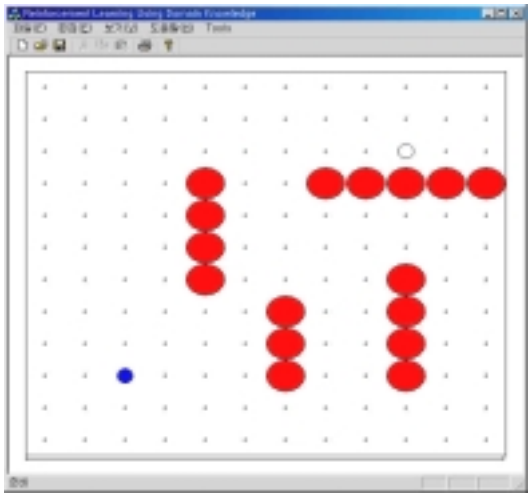

그림6. 12 × 12 map 문제.
Figure6. a new problem of 12 × 12 map.

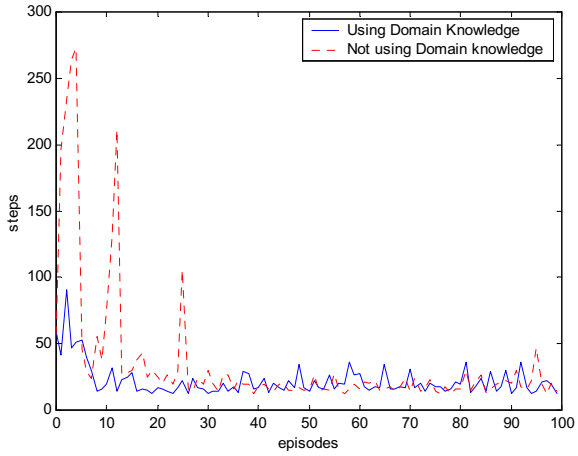

그림7. Q-Learning과 환경지식을 갖는 Q-Learning 비교.
Figure7. Graph of Q-Learning and Q-Learning using Domain Knowledge.

제안한 알고리즘의 유효성을 검증하기 위하여 그림 6과 같은 12 × 12 크기의 Map 문제에 대하여 기존의 Q-Learning과 본 논문에서 제안한 환경지식을 이용한 Q-Learning을 적용한 이동로봇의 모의실험을 통하여 비교하였다. 모의 실험의 결과 그림7에서와 같이 제안한 환경지식을 이용한 학습 알고리즘이 기존의 학습 알고리즘보다 빠른 학습성능을 보이고 있음을 확인할 수 있었다.

6. 결론 및 추후과제

본 논문에서는 강화 학습을 하는 에이전트의 학습속도 개선을 위하여 에이전트가 과거에 학습했던 환경 지식을 새로운 문제에 대한 행위 함수를 학습하는데 이용하는 알고리즘을 자율 이동 로봇에 적용한 모의 실험을 하였고, 환경지식을 안전하고, 적절하게 사용할 수 있도록 본 알고리즘을 확장하여 검증하였다.

에이전트가 환경 지식을 더욱 정교하게 학습하기 위해서는 주어진 하나의 문제에 대한 하나의 Example Set만을 Classifier에 학습시키는 것이 아니라 여러 문제를 총체적으로 학습할 수 있는 능력이 있어야 하며, 이를 위한 연구가 진행되고 있다.

참고문헌

[1] Bryan Singer, Manuela Veloso, “Learning State Features from Policies to Bias Exploration in Reinforcement Learning,” *Technical Note of Carnegie Mellon University*, April, 1999.

[2] C. Watkins, “*Learning from Delayed Rewards*,” PhD Thesis, Cambridge, May, 1989.

[3] C. Watkins, P. Dayan, “Q-learning, technical note,” *Machine Learning*, Vol. 8, pp. 279-292, 1992.

[4] Michael Bowling and Manuela Veloso, “Bounding the suboptimality of reusing subproblems,” *In Proceeding of the NIPS Workshop on Abstraction in Reinforcement Learning*, December, 1998.

[5] R. S. Sutton and A. G. Barto, “*Reinforcement Learning, An Introduction*,” Cambridge, MA : MIT Press, 1998.

[6] 김재현, 서일홍, “지능형 로봇 시스템을 위한 영역기반 Q-Learning,” 제어계측 자동화 로보틱스 연구회 합동학술발표회, pp. 271-276, 1997.