초소형 비행체 운항방법에 대한 환경 지식을 이용한 강화학습 방법

김 봉 오*, 공 성 학*, 장 시 영*, 서 일 홍*, 오 상 록**
*한양대학교 전자전기제어계측공학과, **한국과학기술연구원
(Tel : 82-31-408-5802; Fax : 82-31-408-5803; E-mail : ihsuh@hanyang.ac.kr)

Reinforcement Learning Algorithm using Domain Knowledge for MAV

Bongoh Kim*, Sunghak Kong*, Siyoung Jang*, Il Hong Suh*, Sang-Rok Oh**
*School of Electrical Engineering and Computer Science, Hanyang University
**Korea Institute of Science and Technology
(Tel : 82-31-408-5802; Fax : 82-31-408-5803; E-mail : ihsuh@hanyang.ac.kr)

Abstract - 강화학습이란 에이전트가 알려지지 않은 미지의 환경에서 행위와 보답을 주고받으며, 임의의 상태에서 가장 적절한 행위를 학습하는 방법이다. 만약 강화학습 중에 에이전트가 과거 문제들을 해결하면서 학습한 환경에 대한 지식을 이용할 수 있는 능력이 있다면 새로운 문제를 빠르게 해결할 수 있다. 이런 문제를 풀기 위한 방법으로 에이전트가 과거에 학습한 여러 문제들에 대한 환경 지식(Domain Knowledge)을 Local state feature라는 기억공간에 학습한 후 행위함수를 학습할 때 지식을 활용하는 방법이 연구되었다. 그러나 기존의 연구들은 주로 2차원 공간에 대한 연구가 진행되어 왔다. 본 논문에서는 환경 지식을 이용한 강화학습 알고리즘을 3차원 공간에 대해서도 수행 할 수 있도록 하는 개선된 알고리즘을 제안하였으며, 제안된 알고리즘의 유효성을 검증하기 위해 초소형 비행체의 항공운항 학습에 대해 모의실험을 수행하였다.

1. 서 론

강화학습이란 에이전트(Agent)가 알려지지 않은 환경에서 행동과 보답을 주고받으며, 임의의 상태에서 가장 적합한 행위를 학습하는 방법이다.

Q-Learning은 가장 널리 사용되는 강화학습 방법들 중 하나로 이 학습법은 현재 상태에서의 행위를 미래 행위들로부터 얻게 되는 총 보답을 예측하는 행위 값에 대응시키는 행위함수를 학습하는 알고리즘이다. 그러나 Q-Learning은 빠른 실시간 성능을 가지더라도, 에이전트가 학습을 할 경우 학습 속도가 느리다. 이는 과거에 학습했던 지식을 사용하지 않기 때문이다. 에이전트가 같은 환경 내에서 여러 문제를 해결해야 할 경우, 과거에 문제들을 해결하면서 얻은 환경에 대한 지식을 활용할 수 있다면 학습시간을 줄일 수 있는데, 이러한 연구들로는 Dyna_Q, Policy reuse, Region-based Q-Learning, 그리고 Local state feature to bias exploration등과 같은 연구가 있었다.

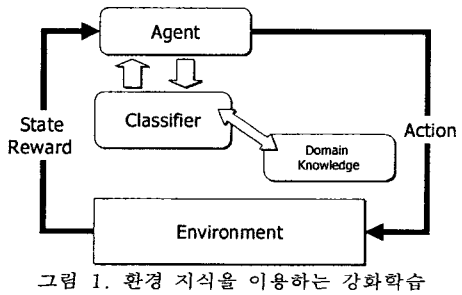

그림 1. 환경 지식을 이용하는 강화학습

Dyna-Q는 가상의 모델을 두어 실제 에이전트가 탐색을 할 때 이용하는 방법이며[5], Policy reuse 방법은 행위함수들의 일부를 재 사용하는 방법이고[4], Region-based Q-Learning은 Q-Learning의 이산 상태 및 이산 행위 공간에서의 학습을 실제 연속 상태 및 연속 행위 공간에 적용하기 위한 연구였다[6].

이러한 알고리즘들 중 Local state feature to bias exploration 방법은 그림 1과 같이 과거에 학습한 여러 문제들에 대한 환경 지식을 Local state feature라는 기억공간에 학습시킨 후, 에이전트가 행위함수를 학습할 때 이 지식을 이용하는 방법이다[1]. 이 알고리즘에 대해서 Local state feature를 더욱 효율적으로 사용하기 위한 개선된 알고리즘에 대한 연구가 자율 이동 로봇에 관해 이루어졌다[7].

지금까지 언급한 방법들은 대부분이 2차원 공간에서 행동을 하는 에이전트에 관한 연구들이었다. 그래서 본 논문에서는 환경 지식을 이용하는 개선된 방법에 대해서 3차원 공간에서 에이전트가 수행 할 수 있도록 하는 방법을 제안하였으며, 제안된 방법의 유효성을 검증하기 위해 초소형 비행체(MAV: Micro Air Vehicle)의 항공운항 학습 방법에 대한 모의실험을 수행하였다.

2. 환경 지식을 이용한 Q-Learning

2.1 Q-Learning

Q-Learning은 대표적인 off-policy 강화학습으로 행위 함수 $Q(s_t, a_t)$의 갱신과 책략 $\pi^*(s_t)$은 다음과 같이 수행된다[2, 3].

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

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

2.2 환경 지식을 이용한 Q-Learning

환경 지식을 이용한 Q-Learning은 Local state feature, Example Set 그리고 Classifier를 가지고 과거의 지식을 이용하여 학습을 하게 된다.

2.2.1 Local state feature

Q-Learning에서 과거에 학습한 행위함수를 새로운 문제에 그대로 적용할 수 없는 이유는 그 행위함수가 과거의 문제에 국한되어 학습이 되어 있기 때문이다. 그래서 주어진 문제와는 독립적인 특성을 갖는 Local state feature라는 저장 공간을 두어 학습에 이용하면 학습을 더욱 효과적으로 할 수 있을 것이다. Local state feature란 에이전트 주위의 상태를 저장하는 공간이다. 예를 들면 에이전트 왼쪽과 오른쪽에 장애물이 있다는 정보를 저장하는 것이다. 이렇게 하여 Local state feature는 주어진 문제와는 독립적이고, 현재 에이전트 상태와 주변 상태들을 관찰함으로써 생성된다.

2.2.2 Example Set

에이전트가 주어진 문제에 대해 행위함수를 학습한 후 환경 지식을 추출하기 위해 Local state feature와 각 상태에서 취한 행위, 그리고 그 행위에 대한 평가가 이루어져야 한다. 이 세 가지의 정보를 갖는 저장 공간을 Example이라고 한다. Example의 생성 시 행위에 대한 평가는 최적의 경로 상에 있는 행위함수의 값 중 가장 작은 값을 문턱 값(Threshold Value)으로 하며, 각 상태에서 취할 수 있는 모든 행위들과 비교하여 문턱 값 보다 큰 경우를 Positive Example 이라고 한다. 이 때, 모든 상태에 대해서 Example을 만들지 않고, 최적 경로 주변의 상태들에 대해서만 Example을 만들어 Example의 수를 줄였다. 그리고 이렇게 생성된 Example들 전부를 Example Set이라 한다.

2.2.3 Classifier

최종적으로 에이전트는 여러 문제들을 통하여 환경 지식을 추출할 때 각 문제를 해결하면서 생성한 Example Set을 이용한다. Classifier는 Example Set의 정보를 가공, 처리하는 역할을 한다. Classifier는 Example Set내의 Example들을 학습하기 위해 Example내의 행위 정보를 참조하고, 행위 정보와 일치하는 Classifier가 학습을 하게 된다. 여기서 Classifier는 Neural Network을 사용하였고, 학습 알고리즘은 Back Propagation Learning Method를 사용하였다. Classifier를 이용한 학습을 효율적으로 하기 위해서는 Classifier내에 저장된 환경 지식 정보의 신뢰도가 높아야 한다. Local state feature는 Goal에 대한 방향에 대한 정보가 없기 때문에, Goal의 방향에 대해 몇 개의 그룹으로 나누고 각 그룹 내에 에이전트 행위에 대한 Classifier를 둔다. 이 때, 각 그룹간의 비율은 가우시안 분포(Gaussian Distribution)를 따른다.

2.2.4 Exploration bias

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

2.1에서 언급한 Q-Learning의 행위 책략 식(2)를 정규화(Normalization)하면 다음과 같은 식이 된다.

$$\pi^*(s_t) = \arg\max_a P_t(s_t,a) \tag{3}$$

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

임의의 상태 s_t에서 행위 a에 대한 Classifier의 값 $C_a(s_t)$을 정규화하면 다음과 같다.

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

따라서 새로운 행위 책략은 다음과 같은 식으로 표현할 수 있다.

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

에이전트가 최적의 행위를 학습하는 과정에서 Classifier의 사용을 초기에는 비중을 많이 두고 학습이 최적에 가까워지는 시기에는 비중을 적게 주기 위해 다음과 같이 식(5)와 식(6)을 변형한다.

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

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

여기서 $Value_{weight}(episode)$는 episode가 증가에 따른 지수함수의 형태를 갖는다.

앞 절에서 언급한 Classifier의 방향에 따른 비율을 고려하기 위해 식(8)을 정리하면 다음과 같게 된다.

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

$$P'(s_t,a) = \sum_i P_i(s_t,a)\cdot W_i, \quad i = 0°,45°,\mathrm{K},315° \tag{9}$$

$$W_i(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2}\left(\frac{x-i}{\sigma}\right)^2\right) \quad x: angle$$

여기서 $W_i(x)$는 방향에 따른 Classifier의 비율을 나타낸다.

3. 3차원 공간의 환경 지식을 이용한 Q-Learning

환경 지식을 이용한 Q-Learning을 3차원 공간에 적용하기 위해서는 3차원 행위 공간에 대한 Classifier의 구성과 각 Classifier의 비율에 대한 수정이 필요하다.

3.1 행위 공간

3차원 공간에서 상태를 discrete 하게 나누어 보면, 그림 2에서 보듯이 하나의 에이전트가 취할 수 있는 행위는 26개가된다.

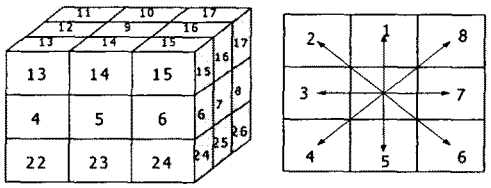

그림 2. 3차원 공간에서의 행위 공간

3.2 Classifier

앞 절에서 언급했듯이 에이전트와 Goal과의 방향에 대해서 고려하기 위해, 3차원 공간에서 방향에 대한 그룹을 3차원 공간에서 행위 방향들과 같게 하여 26개의 Classifier의 그룹을 만들고, 각 그룹마다 Classifier를 행위 개수만큼을 둔다.

3차원 공간에서도 역시 Classifier의 방향에 따른 비율을 고려하여야 한다. 식(9)에서 보듯이 2차원 공간의 경우에는 방향에 따른 비율을 Goal에 대한 방향 각도의 함수인 가우시안 분포로 표현할 수 있었다.

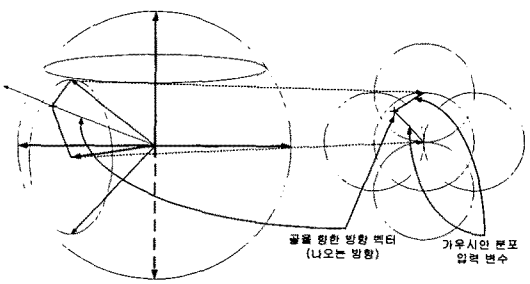

그림 3. 3차원 공간에서의 가우시안 분포 입력 변수

그러나 3차원 공간에서는 2차원 공간에서와 같이 Goal과의 방향각도 만을 가지고 비율을 계산하기가 어렵다. 그래서 그림 3과 같이 반지름이 1인 가상의 구를 만들고, 구의 표면에 에이전트의 행위 방향인 26개의 점들을 만드는데, 이 점들은 Goal의 방향에 따른 26개의 Classifier 그룹들을 대표하게 된다. 가우시안 분포의 입력 변수로는 Goal을 향하는 방향벡터와 26개의 구 표면의 점들과의 거리를 취하게 된다. 따라서 식(9)를 정리하면 다음과 같이 된다.

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

$$P'(s_t,a) = \sum_i P_i(s_t,a)\cdot W_i, \quad i = 1,2,\ldots,26 \qquad (10)$$

$$W_i(x) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{1}{2}\left(\frac{x-i}{\sigma}\right)^2\right) \quad x : \text{distance}$$

그림 4는 거리에 따른 가우시안 분포 $W_i(x)$를 나타낸다.

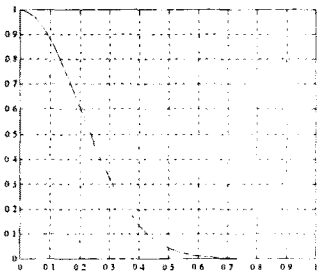

그림 4. 거리에 따른 가우시안 분포

4. 모 의 실 험

본 논문에서 제안된 3차원 공간에서의 환경 지식을 이용한 강화학습을 적용하여 실험을 하기 위해서, 에이전트를 초소형 비행체로 하고 그림 5와 같은 실험 환경을 구성하였다. 그리고 환경 지식을 이용한 Q-Learning과 기존의 Q-Learning의 학습 성능에 대한 실험을 비교 분석하였다.

에이전트의 3차원 공간에서 가능한 행위는 26개의 방향으로 정하였고, 그림 5와 같은 유형의 문제를 각 행위 방향마다 30개씩, 총 780개의 문제를 이용하여 에이전트의 환경 지식을 미리 학습을 시켰다.

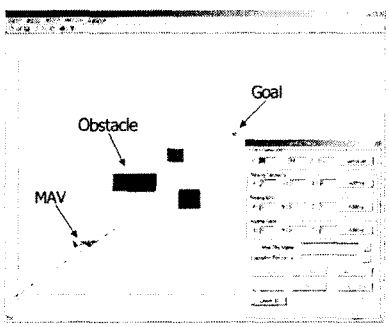

그림 5. 10×10×10 map 문제

Classifier의 그룹도 역시 행위 방향의 수와 같게 두었다. Local state feature는 임의의 상태에서 거리가 4인 상태들을 관찰하여 구성하였으며, Classifier에 사용한 Neural Network의 구조는 하나의 숨은 계층(Hidden layer)을 갖는 구조로 하였으며, 입력 계층, 숨은 계층, 그리고, 출력 계층내의 뉴런의 개수는 각각 257개, 20개, 그리고 1개로 구성하였다.

제안한 3차원 공간에 대한 환경 지식을 이용한 Q-Learning 알고리즘을 검증하기 위해서, 그림 5와 같은 유형의 문제를 10×10×10크기의 map 36개와 15×15×15크기의 map 28개, 총 64개를 이용하여 기존의 Q-Learning과 본 논문에서 제안된 환경 지식을 이용한 Q-Learning에 대해서 초소형 비행체의 모의실험을 통하여 비교하였다.

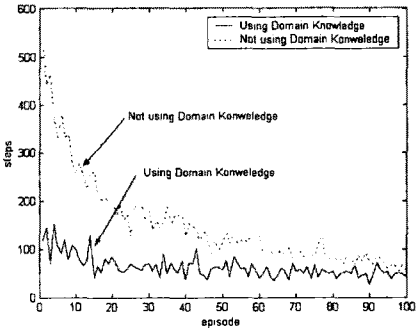

그림 6. Q-Learning과 환경 지식을 이용한 Q-Learning 비교

모의실험 결과 그림 6과 같이 제안한 환경 지식을 이용한 Q-Learning이 기존의 학습 알고리즘 보다 빠른 학습 성능을 보이고 있음을 확인할 수 있었다.

5. 결 론

본 논문에서는 에이전트가 새로운 문제를 학습할 때 과거에 학습했던 환경지식을 이용하는 알고리즘을 3차원 공간상에서도 적용할 수 있도록 개선하였으며, 제안한 알고리즘을 3차원 공간상에서 자율 비행하는 초소형 비행체의 항공운항 학습에 관한 모의실험을 통해 알고리즘의 유효성을 검증하였다.

실험 결과를 보면 환경 지식을 이용한 Q-Learning이 기존의 Q-Learning보다 초기의 학습 속도가 빠른 것을 볼 수 있었다.

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 하겠다.

[참 고 문 헌]

[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", 제어 자동화 시스템 공학회지, Vol.3, No. 4, pp. 420-425, 1997.

[7] 장시영, 공성학, 서일홍, 오상록, "Domain Knowledge를 이용한 강화학습", Proc. of the International Conference on Control Automation and Systems, October, 2001.