Q-value Initialization을 이용한 Reinforcement Learning Speedup Method

최 정 환
Dept. of Electrical and Computer Engineering, Purdue University
전화 : 031-973-9665

Reinforcement learning Speedup method using Q-value Initialization

Juang Hwan Choi
Dept. of Electrical and Computer Engineering, Purdue University
E-mail : jchoi@ecn.purdue.edu

Abstract

In reinforcement learning, Q-learning converges quite slowly to a good policy. Its because searching for the goal state takes very long time in a large stochastic domain. So I propose the speedup method using the Q-value initialization for model-free reinforcement learning. In the speedup method, it learns a naïve model of a domain and makes boundaries around the goal state. By using these boundaries, it assigns the initial Q-values to the state-action pairs and does Q-learning with the initial Q-values. The initial Q-values guide the agent to the goal state in the early states of learning, so that Q-learning updates Q-values efficiently. Therefore it saves exploration time to search for the goal state and has better performance than Q-learning. I present Speedup Q-learning algorithm to implement the speedup method. This algorithm is evaluated in a grid-world domain and compared to Q-learning.

I. 서론

Reinforcement learning은 Stochastic 도메인에서 Policy를 찾는 매우 유용한 방법이다. Q-learning은 이러한 Reinforcement learning 알고리즘중 가장 널리 쓰이고, 가장 효율적인 model-free 알고리즘이다.[2] Q-learning 알고리즘은 각 state-action pair의 Q-value, 즉 Q(s,a)을 학습한다. Q(s,a)는 state s에서 action a를 행하고, 이후의 state들에서 최적화된 action을 행했을 때, 얻을 수 있는 reward들의 기대값이다. 그러나, 비록 Q-learning이 가장 효율적인 model-free Reinforcement learning 알고리즘이라 하여도, 이 알고리즘의 Q-value 값은 매우 느리게 최적치로 수렴한다. 이는 이론적으로 Q-value가 Stochastic 도메인에서 수렴하기 위해서는, Q-learning 알고리즘이 무한히 Q-value를 업데이트하여야 하기 때문이다.[3]

이러한 기본적인 문제점 외에도 Q-learning 알고리즘은 그 학습의 초반기에 매우 나쁜 학습효율을 보인다. 이는 Q-learning 알고리즘이 학습 초반기에 도메인을 explore하여 goal state를 찾아야 하는데, 이때 action을 랜덤하게 선택하기 때문이다. 아무런 가이드 없이 랜덤한 action으로 goal state를 찾는데는 매우 오랜 시간이 걸리며, 이것이 학습 초반기에 낮은 학습효율을 보이게 하는 주 원인이다. 이전의 연구에서는 수퍼바이저가 initial policy를 learning system에 제공

함으로서, Q-learning 알고리즘이 학습 초반기에 능률적으로 동작하도록 했다.[4] 그러나, 이 경우 학습하려는 도메인의 전문가가 필요하다.

본 논문에서는 도메인 전문가의 간섭을 배제한, model-free Reinforcement learning 알고리즘을 위한 Speedup method 연구를 수행하려 한다. 기본적으로 Model-free reinforcement learning 알고리즘은 학습중에 모은 트랜지션 데이터를 효율적으로 사용하지 못한다.[2] Q-learning 알고리즘의 경우 한 state에서 어떠한 action을 통하여 다른 state로 옮겨가는 트랜지션 데이터를 단지 각 state-action pair의 Q-value 업데이트에만 사용하므로, 학습 초반기의 경우, 이 트랜지션 데이터가 효율적으로 사용되지 못한다.

본 논문에서는 model-free reinforcement learning을 위한 speedup method를 제안한다. 본 논문에서는 도메인의 Naive model을 이용하여, 각 state의 shortest path를 구하고, 얻어진 shortest path를 이용하여 각 state-action pair의 initial Q-value를 구한다. 모든 state-action pair의 initial Q-value가 구해지면, 얻어진 initial Q-value를 가지고 Q-learning을 실시한다. Q-learning이 끝나면 업데이트된 Q-value값을 가지고 Greedy strategy를 이용함으로서 Policy를 구할 수 있다. 본 Speedup method에서는 Q-learning과는 달리 initial Q-value를 이용하여 학습을 시작하므로, 주어진 Q-value들이 Agent를 goal state로 인도한다. 따라서 학습 초반기에 능률적인 학습이 가능하여지고, 결과적으로 같은 시간 안에 Q-learning 보다 나은 성능의 Policy를 구할 수 있다.

II. Speedup Methods

2.1 Assumptions

Speedup method가 쓰이는 도메인은 다음과 같은 조건을 가진다.

(1) 도메인의 state 와 action space는 discrete 하고 finite 하다.

(2) 도메인은 stochastic 도메인이다.

(3) Goal state가 존재한다.

2.2 Q-value initialization을 이용한 speedup method.

Speedup method는 크게 두 부분으로 나뉜다. 첫 번째 파트에서 stochastic 도메인의 naive 모델을 구하고 이 모델을 이용하여 각 state에서 Goal state로의 shortest path를 구한다. naive 모델은 하나의 state-action pair로부터 트랜지션이 가능한 다음 state들 중 하나의 state로의 deterministic mapping 이다. 본 논문에서는 naive model을 표현하기 위하여 Next(s,a)를 사용한다. 여기서 구한 naive 모델은 단지 shortest path를 구하는데만 쓰여지고, Q-value function을 찾는 데는 쓰여지지 않는다. 이렇게 구하여진 shortest path는 initial Q-value를 구하는데 쓰여지고, 이 Q-value는 이후에 Q-learning을 통하여 업데이트 되므로, 정확한 stochastic 모델이 아닌 naive 모델로도 충분하다.

naive 모델을 구하기 위하여 우선 $Next(s_0,a_0)$가 아직 기록되지 않은 state s_0와 action a_0를 선택한다. action a_0를 실행하고, 다음 state s_1를 관찰하여, $Next(s_0,a_0)=s_1$를 기록한다. 만일 s_1이 아직 기록되지 않은 $Next(s_1,a_1)$을 가지고 있는 a_1을 가지고 있으면 a_1을 실행시키고 다시 action a_2를 찾는다. 만일 a_1이 존재하지 않으면 다시 s_0를 구하고 실행, 기록을 반복한다.

naive 모델을 구한 후에 이를 이용하여 각 state에서 Goal state까지의 shortest path를 찾는다. 본 논문에서는 각 state의 reward가 같다는 가정 하에 boundary 개념을 이용하여 shortest path를 구한다. $Boundary_i$는 state들의 set으로 $Boundary_i$에 포함되는 각 state s의 Next(s,a)는 $Boundary_{i-1}$에 포함되며. $Boundary_0$는 Goal state들의 set이다.

$Boundary_0$는 Goal state들의 set이므로, Next(s,a)가 $Boundary_0$에 포함되는 state들은 $Boundary_1$에 포함되고, 이 state들은 Goal state 들로부터 1 step 떨어져 있는 state들이다. 따라서 이런 식으로 Boundary들을 구하게 되면, $Boundary_i$는 Goal state로부터 i step 떨어진 state들의 set이 된다.

본 논문에서 실제 stochastic 모델이 아닌 naive 모델, 즉 deterministic 모델을 이용하므로 worst case의 경우 boundary를 구하지 못할 수도 있다. 이 경우 initial Q-value 값을 구할 수 없다. 그러나 worst case의 경우, Speedup method는 Q-learning과 똑같은 상황에서 학습을 시작하게 되므로, Q-learning과 똑같은 성능을 가지게 된다.

Speedup method의 두 번째 파트는 주어진 shortest path를 이용하여 initial Q-value를 구하고 Q-learning을 하는 것이다. initial Q-value는 간단히 구할 수 있다. state s가 $Boundary_i$에 포함되고, Next(s,a)가 $Boundary_{i-1}$에 포함될 경우 Q(s,a)는 i 이다.

모든 state의 initial Q-value를 구한 후, 그 값을 가지고 Q-learning을 하면 policy를 구할 수 있다.

Speedup method는 initial Q-value가 Goal state의 위치를 대략적으로 알려주므로, 학습의 초반에 exploration time을 아낄 수 있어, 같은 학습 시간이 주어질 경우, 더 나은 성능의 policy를 구할 수 있다. 또한 worst case의 경우에도 Q-learning과 똑같은 성능을 가지므로, Speedup method는 언제나 Q-learning보다 같거나 나은 성능을 보인다.

2.3 Speedup Q-learning

Speedup Q-learning은 speedup method를 구현한 알고리즘이다. 본 알고리즘에서 Goal state는 absorbing state 이며, reward는 0이라고 가정했으며, 다른 state들의 reward는 1이라고 가정했다. state의 수는 m이며, action의 수는 n이다. 본 알고리즘은 두 개의 파라메터를 가진다. *life time*은 학습하는 전체 시간이며, γ는 Q-value update rule의 discount factor이다.

Speedup Q-learning (*life time*, γ)

(1) Initially G={ }, Q(s,a)=∞, and Next(s,a)=null for all s and a.
  While (there is Next(s,a)=null)
    Pick a state s randomly
    While (there is an action a whose Next(s,a)=null)
      Select an action a whose Next(s,a)=null and execute it
      Observe the new state s' and Next(s,a)=s'
      If (s is the goal state)
        G=G∪{s},
        Q(s,a)=0 and Next(s,a)=s for all a
      s=s'

(2) Initially $Boundary_i$ ={ }, $Boundary_0$=G, and Domain={all states}-G
  i=0
  While (Domain is not empty)
    For (j=1...number of elements in Domain)
      Pick a state s of Domain
      If (there is Next(s,a) which is included in $Boundary_i$)
        $Boundary_{i+1}$=$Boundary_{i+1}$∪{s}
        Domain=Domain-{s}
    If ($Boundary_{i+1}$={ })
      $Bounday_{i-1}$= Domain
      Domain={ }
    i=i+1

(3) *For (i=1...m)*
    For (j=1...n)
      If (s_i∈$Boundary_k$, Next(s_i,a_j)∈$Boundary_{k-1}$)
        $Q(s_i,a_j)$=k

(4) Q-learning

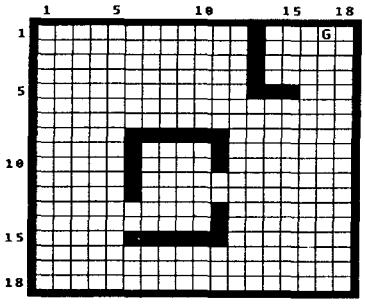

그림 1. grid-world 도메인

III. Experiments

본 논문에서는 Q-learning과 Speedup method의 성능을 비교하기 위하여 Grid-world 도메인을 이용한다. 같은 도메인에서 각 알고리즘을 이용하여 각각의 Policy를 구하고, 성능을 측정한다. 본 논문의 경우 각 state에서 goal state까지 도달하는 평균 스텝수를 측정하여 비교하였고, Speedup method가 더 좋은 성능을 가짐을 보였다

그림 1은 18×18 grid-world 도메인이며, 본 도메인의 테스크는 각 state에서 Goal state에 도달하기 위한 optimal policy를 찾는 것이다. 본 도메인은 297개의 state를 가지고 있고, 각 state는 4개의 action(up, down, left, right)을 가지고 있다. 본 도메인은 stochastic 도메인이므로, 각 160개의 action들은 20%, 50%, 75%, 95%의 성공률을 가지고, 나머지 action들은 100%의 성공률을 가진다. 각 state의 reward는 1이고, 그림에서 G로 표시되는 Goal state의 reward는 0이다. 학습을 시작하기전에 각 state의 Q-value 값은 200이다.

3.1 Experiments of Q-learning and Speedup Q-learning

실시한 모든 Q-learning의 실험에서 discount factor는 1이고, simulated annealing exploration strategy가 사용되었다. 학습이 끝난 후, 각 state에서 policy를 따라 움직였을 때 Goal state 까지 도달하는데 걸린 step 수를 측정하고 평균을 구했다.

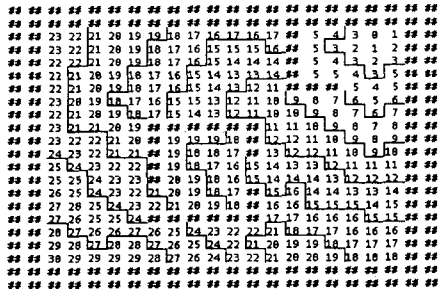

그림 2. boundaries

Speedup Q-learning의 실험에서 모든 조건은 Q-learning의 실험과 동일하며, Speedup Q-learning에 포함되어 있는 Q-learning은 위의 Q-learning과 동일한 것이다. Speedup Q-learning의 (2)가 끝난후 모든 state들은 각자의 boundary에 속하게 되며, 그림 2는 실험중에 구한 boundary들을 보여준다. 각 grid, 즉 state의 번호는 각 state가 속한 boundary의 번호를 나타낸다.

두 알고리즘을 비교하기 위해, 각각의 life time에서 10회의 실험을 하였고, 각 실험 후에 Goal state까지의 평균 step수를 측정하였다. 그림 3은 10회 실험의 평균값이며, Speedup Q-learning이 Q-learning보다 더 빠르게 수렴함을 보여준다.

?.2 Experiments with a more stochastic dynamics

본 실험에서는 위의 실험과는 다른 dynamics를 가진 도메인에서 같은 실험을 하였다. 본 도메인에서는 전체 action 중, 각각 1/4개의 action들이 20%, 50%, 75%, 95%의 성공률을 가진다. 나머지 조건들은 위의 실험에 쓰인 grid-world 도메인과 동일하다.

그림 4는 Speedup Q-learning이 Q-learning 보다 더 빠르게 수렴함을 보여준다.

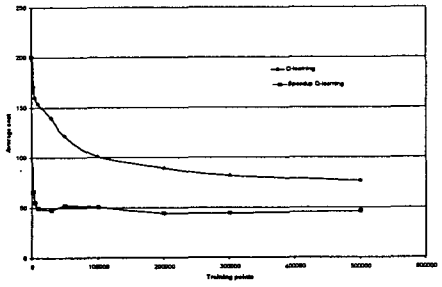

그림 3. Average step number

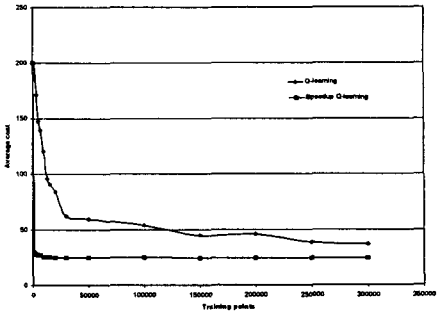

그림 4. Average step number

Ⅳ. 결론

본 논문에서는 model-free reinforcement learning을 위한 speedup method를 제안하였다. speedup method는 naive 모델을 이용하여 각 state-action pair의 initial Q-value 값을 구하고, 이 값을 이용하여 Q-learning을 한다. initial Q-value 값이 learning system에 Goal state까지의 대략적인 방향을 제시하므로 learning system은 learning의 초반부에서 exploration time을 아낄 수 있다. 따라서 같은시간내에 Q-learning보다 더 나은 policy를 구할 수 있다.

참고문헌

[1] T.G. Dietterich and N.S. Flann, Explanation -Based Learning and Reinforcement Learning: A Unified View, *Proceedings of the 12the International Conference on Machine Learning*, pp. 176-184, 1995.

[2] L.P. Kaelbling, M.L. Littman and A.W. Moore, Reinforcement Learning: A Survey, *Journal of Artificial Intelligence Research*, 4, 237-285, 1996.

[3] T. Mitchell, *Machine Learning*, McGraw-Hill, 1998.

[4] W.D. Smart and L.P. Kaelbling, Practical Reinforcement Learning in Continuous Spaces, *Proceedings of the 17th Int. Conference on Machine Learning*, 2000.

[5] R.S. Sutton, Integrated architectures for learning, planning, and reacting based on approximating dynamic programming, *Proceedings of the 7th Int. Conference on Machine Learning*, 1990.