복잡계망 모델을 사용한 강화 학습 상태 공간의 효율적인 근사

(Efficient Approximation of State Space for Reinforcement Learning Using Complex Network Models)

이 승 준 † 엄 재 홍 † 장 병 탁 ††
(Seung-Joon Yi) (Jae-Hong Eom) (Byoung-Tak Zhang)

요 약 여러 가지 실세계 문제들은 마르코프 결정 문제(Markov decision problem) 들로 형식화하여 풀 수 있으나, 풀이 과정의 높은 계산 복잡도 때문에 실세계 문제들을 직접적으로 다루는 데 많은 어려움이 있다. 이를 해결하기 위해 많은 시간적 추상화(Temporal abstraction) 방법들이 제안되어 왔고 이를 자동화하기 위한 여러 방법들 또한 연구되어 왔으나, 이들 방법들은 명시적인 효율성 척도를 갖고 있지 않아 이론적인 성능 보장을 하지 못하는 문제가 있었다. 본 연구에서는 문제의 크기가 커지더라도 좋은 성능이 보장되는 자동적인 시간적 추상화 구현 방법에 대해 제안한다. 이를 위하여 네트워크 척도(Network measurements)를 이용하여 마르코프 결정 문제의 풀이 효율과 상태 궤적 그래프(State trajectory graph)의 위상 특성간의 관계를 분석하고, 네트워크 척도들 중 평균 측지 거리(Mean geodesic distance)가 마르코프 결정 문제의 풀이 성능과 밀접한 관계가 있다는 사실을 알아내었다. 이 사실을 기반으로 하여, 낮은 평균 측지 거리를 보장하는 복잡계망 모델(Complex network model)을 사용하여 시간적 추상화를 만들어 나가는 알고리즘을 제안한다. 제안된 알고리즘은 사실적인 3차원 게임 환경을 비롯한 여러 문제에 대해 테스트되었고, 문제 크기의 증가에도 불구하고 효율적인 풀이 성능을 보여 주었다.

키워드 : 강화 학습, 시간적 추상화, 성능 척도, 위상 특성, 복잡계망 모델, 평균 측지 거리

Abstract A number of temporal abstraction approaches have been suggested so far to handle the high computational complexity of Markov decision problems (MDPs). Although the structure of temporal abstraction can significantly affect the efficiency of solving the MDP, to our knowledge none of current temporal abstraction approaches explicitly consider the relationship between topology and efficiency. In this paper, we first show that a topological measurement from complex network literature, mean geodesic distance, can reflect the efficiency of solving MDP. Based on this, we build an incremental method to systematically build temporal abstractions using a network model that guarantees a small mean geodesic distance. We test our algorithm on a realistic 3D game environment, and experimental results show that our model has subpolynomial growth of mean geodesic distance according to problem size, which enables efficient solving of resulting MDP.

Key words : Reinforcement Learning, Temporal abstraction, Measurement of efficiency, Topological property, Complex network model, Mean geodesic distance

† 학생회원 : 서울대학교 전기컴퓨터공학부
sjlee@bi.snu.ac.kr
jheom@bi.snu.ac.kr
†† 종신회원 : 서울대학교 전기컴퓨터공학부 교수
btzhang@bi.snu.ac.kr
논문접수 : 2008년 11월 19일
심사완료 : 2009년 4월 1일

정보과학회논문지: 소프트웨어 및 응용 제36권 제6호(2009.6)

1. 서 론

1.1 서론

다양한 실세계 문제들이 마르코프 결정 문제(MDP; Markov decision problem)로 모델링될 수 있다. 이러한 MDP들을 풀기 위해서 평가치 반복(Value iteration)이나 정책 반복(Policy iteration) 등의 알고리즘 등이 제안되어 왔으며, MDP의 확률 모델에 대해 정보가 없는 보다 일반적인 경우 강화 학습(Reinforcement learning)이 사용되고 있다. 하지만 이들 알고리즘들은 대부

분 이산적인 상태 공간(State space)들을 가정하고 있고, MDP의 크기가 커질 경우 적용하기 힘든 어려움이 있어, 크기가 크거나 연속적인 상태 공간을 가지는 실세계 문제들에 적용하기엔 어려움이 많다. 보편적인 해법으로는 신경망(Neural network) 과도 같은 함수 근사기(Function approximator)를 사용하여 상태 공간을 근사하는 방법이 쓰이고 있으나, 이 경우에도 상태 공간을 근사하는 복잡도가 증가함에 따라 속칭 '차원성의 저주'(Curse of dimensionality)로 불리는 학습해야 할 파라미터 수의 폭발적인 증가가 나타난다는 것이 알려져 있다[1].

이러한 차원성의 저주를 해결하기 위한 방법으로 제안된 것이 각각의 단위 행동(Action) 마다 판단을 하지 않고, 여러 단계를 묶어서 판단을 하는 시간적 추상화(Temporal abstraction) 방법들이다[2]. 이러한 방법들은 계층적인 제어 구조를 이루게 되고, 이러한 계층적인 구조를 사용하는 강화 학습을 계층적 강화 학습(Hierarchical reinforcement learning)이라 칭한다. 이러한 시간적 추상화 방법을 사용할 경우 MDP의 학습이 크게 빨라짐이 기존 여러 연구를 통해 실험적으로 보여왔다.

하지만 현재까지 제안된 여러 시간적 추상화 방법들은 다음과 같은 몇 가지 문제점을 가지고 있다. 우선 대부분의 방법들은 사용자가 문제에 대한 지식을 바탕으로 사전에 계층 구조를 디자인할 것을 요구하지만, 실세계 문제들의 경우 문제의 구조를 사전에 알기 힘든 경우가 많다. 또한 현재의 시간적 추상화 방법들은 명시적 효율성 척도를 가지고 있지 않아, 사용자가 디자인하거나 자동으로 디자인된 계층 구조가 얼마나 효율적인지에 대한 보장이 되지 않는다. 마지막으로 현재의 방법들은 대부분 이산적인 상태 공간과 테이블 형태의 평가치 저장을 사용하기 때문에 연속적인 상태 공간을 갖는 실세계 문제에 바로 적용하기가 어렵다. 첫 번째 문제를 해결하기 위해 자동적으로 계층 구조를 디자인하려는 여러 시도가 있어 왔으나, 이 방법 역시 다른 문제점들을 여전히 가지고 있다.

이러한 문제점들을 해결하기 위해, 두 가지 질문에 대답할 필요가 있다. MDP를 푸는 효율을 어떻게 측정할 것인가, 그리고 연속된 상태 공간을 갖는 문제에 대해, 효율적인 풀이가 보장되는 MDP 모델을 어떻게 만들 것인가가 그 질문들이다. 첫 번째 질문에 대한 우리의 가설은 MDP를 푸는 효율은 그 MDP의 위상학적 성질과 밀접한 관계가 있으며, 이는 네트워크 척도(Network measurements)들을 사용하여 측정 가능할 거라는 것이다. 본 연구에서는 이를 보이기 위해 다양한 구조와 크기의 MDP에 대해 대응되는 상태 경로 그래프(State trajectory graph)를 만들고, 여러 네트워크 척도들을 실험적으로 측정하여 네트워크 척도들 중 하나인 평균 측지 거리(Mean geodesic distance)가 MDP의 풀이 효율을 나타낼 수 있는 척도로 사용 가능하단 것을 보였다. 두 번째 질문에 답하기 위해, 연속된 상태 공간을 갖는 MDP를 근사할 수 있는 그래프 기반 상태 근사 장치[3]를 사용하여 MDP를 근사하고, 다시 여기에 짧은 평균 측지 거리를 보장해 주는 복잡계망 모델(Complex network model)을 사용하여 이를 낮은 평균 측지 거리를 갖도록 하였다. 실험적인 결과를 통해서 제안된 모델은 문제의 복잡도가 증가하더라도 평균 측지 거리가 크게 증가하지 않으며, 결과적으로 우수한 풀이 효율을 유지할 수 있음을 확인할 수 있었다.

1.2 관련 연구

1.2.1 마르코프 결정 문제

마르코프 결정 프로세스(Markov decision process) M은 순서쌍 $<S,A,T,R>$으로 주어진다[4]. S는 상태들 s의 유한집합이고, A는 행동들 a의 유한집합, R은 보상(Reward) 을 정의하는 함수 $r(s,a,s')$이고, T는 환경이 상태 s로부터 s'로 이동할 확률인 $p(s'|s,a)$로 주어지는 상태 변화 함수(State transition function)이다. 정책 π는 대응 $\pi : S \rightarrow A$로 정의된다. 마르코프 결정 문제는 주어진 정책으로부터 각 상태를 평가하게 해 주는 성능 지표가 마르코프 결정 프로세스에 추가된 것이며, MDP 이론의 원론적인 결과는 모든 MDP들은 모든 상태에 대해 각 상태의 평가 함수(Value function) $V^{\pi}(s)$를 최대화시키는 정책인 최적 정책 π^{*}를 가진다는 것이다. 이 π^{*}를 구하는 것이 MDP를 푸는 것이며, MDP를 풀기 위해 널리 쓰이는 반복적인 알고리즘으로는 각각 다음의 갱신 규칙을 사용하는 평가치 반복 알고리즘과 Q-학습이 있다[5].

$$V(s) \leftarrow \max_{a} \sum_{s'} p(s'|s,a)[r(s,a,s') + \gamma V(s')] \quad (1)$$

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma^{k(s,a)} \max_{a'} Q(s',a') - Q(s,a)] \quad (2)$$

일반적으로 모델이 주어졌을 경우, MDP 풀이 알고리즘들의 계산 복잡도는 문제의 크기에 다항식적 비례 축소(Polynomial scaling)를 보임이 알려져 있으나[6], 모델이 알려져 있지 않은 경우 강화 학습 알고리즘들은 문제의 크기에 지수적인 비례 축소를 보일 수도 있다는 것 또한 알려져 있다[7]. 따라서 문제의 크기가 크거나 연속적인 실세계 문제들을 직접적으로 MDP로 형식화하여 푸는 데는 어려움이 많다.

1.2.2 시간적 추상화

큰 MDP를 효율적으로 풀기 위한 대표적인 방법은 하위 작업(Subtask)이나 다단계 행동(Multi-step action)을 사용하여 두 상태들 간에 필요한 의사 결정 단계를

줄이려 하는 시간적 추상화 방법이다[1]. 계층적(Hierarchical) 방법에서는 MDP가 각 단계마다 여러 개의 하위 MDP로 나뉘게 되고, 각 하위 MDP는 상위 MDP의 정책을 하위 목표(Subgoal)로 사용하게 된다[8]. 선택권(Option) 체계의 경우 MDP가 계층적으로 조직화되는 대신, 각 상태의 행동들에 다단계 행동을 한 번에 선택 가능한 확장된 행동인 선택권이 추가되게 된다[2]. 이들 시간적 추상화 방법들은 각 행동들을 행하는 데 단위 시간이 아닌 다양한 시간 $t(s,a)$이 걸릴 수 있도록 확장된 MDP 모델인 세미-MDP(Semi-MDP) 모델에 의해 형식화될 수 있다. 이러한 세미-MDP를 푸는 데는 앞서 다룬 평가치 반복이나 Q-학습 알고리즘의 갱신 규칙을 다음과 같이 약간의 수정을 거쳐 그대로 사용 가능하다.

$$V(s) \leftarrow -\max_a \sum_{s'} p(s'|s,a)[r(s,a,s') + \gamma^{t(s,a)} V(s')] \quad (3)$$

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma^{t(s,a)} \max_{a'} Q(s',a') - Q(s,a)] \quad (4)$$

이러한 시간적 추상화를 사용할 경우, 시간적 추상화의 위계 구조를 MDP 풀이 과정에서 명시적으로 사용하지 않고 기존 알고리즘을 그대로 사용하더라도 MDP의 풀이 효율이 크게 향상된다는 것이 실험적으로 알려져 있다. 즉, 시간적 추상화를 통해 MDP가 일반적으로 더 풀기 효율적인 형태로 변화되었다고 볼 수 있다.

1.2.3 자동적 시간적 추상화 방법들

시간적 추상화 방법들의 가장 큰 단점은 사용자가 사전에 계층 구조를 제공해야 한다는 것이고, 이 점을 해결하기 위해 자동적으로 이를 만들려는 많은 시도가 있어 왔다. 이들 시도들은 다음과 같이 몇 가지로 크게 구분될 수 있다. 우선 유사성 기반 방법들은 학습된 정책들 사이에서 유사성을 찾아 이들로부터 재사용 가능한 다단계 행동이나 정책을 만들어 내는 방법들이다[9]. 반면 특이성 기반 방법들은 보상 함수로부터 통로의 갈림길, 방의 입구 등 특이한 상태들을 찾아내어 이들을 하위 목표로 사용하는 방법이다[10]. 빈도 기반 방법들은 마찬가지로 특이한 상태들을 찾아내어 하위 목표로 사용하는 방법들로, 각 상태들의 방문 빈도를 바탕으로 특이한 상태들을 찾아낸다[11]. 또한 구분 기반 방법들은 하나의 큰 상태로부터 출발하여 충분한 상이성이 관측될 경우 상태를 나눠 나가는 방법이다[12]. 마지막으로 그래프 기반 방법들은 MDP로부터 상태 궤적 그래프를 만든 뒤 이 그래프에 그래프 이론적인 방법들을 사용하여 하위 목표를 찾아내는 방법이다[13].

하지만 이러한 자동적 시간적 추상화 방법들은 다른 여러 단점들을 여전히 가지고 있다. 즉 기본적인 강화 학습에서 사용하는 이산적인 상태 공간과 테이블 형태의 평가치 저장 방법을 여전히 사용하고 있기 때문에 연속적이거나 규모가 큰 실제 문제에 직접적인 적용이 쉽지 않으며, 다양한 방식으로 계층 구조를 자동적으로 형성하지만 거기에 대한 이론적인 기반이 존재하지 않아 생성된 계층 구조의 성능에 대한 보장이 되지 않는다는 보다 큰 문제를 가지고 있다.

본 연구에서 제안하는 방법은 기존 방법들 중에서 크게 그래프 기반 방법에 속한다고 볼 수 있으나, 계층 구조와 성능간의 관계를 측정할 수 있는 구체적인 성능 척도를 사용하고 그 척도에 기반하여 결과의 성능이 보장되는 계층 구조를 생성하며, 처음부터 연속적인 문제를 고려하였다는 등의 점에서 기존 방법들과 크게 차별된다고 볼 수 있다.

2. MDP와 네트워크 척도(Network measurements)

2.1 MDP의 위상 구조와 풀이 효율

앞서 살펴보았듯이 다양한 방법으로 MDP에 시간적 추상화를 추가할 수 있고, 이는 일반적으로 MDP의 풀이 효율을 향상시키게 된다. 하지만 현재의 방법들은 명시적인 성능 지표를 갖고 있지 못하기 때문에, 우리는 이러한 시간적 추상화의 추가가 MDP의 풀이 효율을 어느 정도 향상시킬지에 대해 전혀 알 수 없고, 문제가 커짐에 따라 풀이 효율이 어떻게 달라질 지에 대해서도 알 수 없다. 시간적 추상화의 위상 구조에 따라 풀이 효율이 크게 달라질 수 있다는 것은 다음 예를 통해 알아볼 수 있다.

그림 1은 세 가지의 MDP 구조에 대해 평가치 반복 알고리즘을 적용하였을 경우의 문제 크기에 따른 풀이 시간의 변화를 나타낸 그래프이다. 그림 1(a)의 기본적인 MDP에 대해 2단계 행동인 선택권을 추가한 그림 1(b) 구조의 경우, 풀이 시간은 절반으로 줄어들었으나 문제 사이즈가 변화함에 따른 풀이 시간의 변화는 동일한 다항식적 비례 축소(Polynomial scaling) 양상을 보인다. 반면 계층적 구조를 추가한 그림 1(c)의 경우 전혀 다른, 대수 다항식적 비례 축소(Polylogarithmic scaling) 양상을 보인다. 여기서 알 수 있는 것은 계층 구조를 추가할 경우 풀이 효율이 증가하지만, 큰 문제에 대해서 우수한 풀이 효율을 반드시 보장할 수는 없다는 것이다. 이를 위해서는 풀이 효율과 밀접하게 관련된 명시적인 척도(Measurement)가 필요하다. 이는 아래에서 좀 더 자세히 알아보도록 한다.

2.2 MDP와 상태 궤적 그래프

위에서 시간적 추상화가 MDP를 더 풀기 효율적인 형태로 바꿔 주지만, 명시적인 성능 지표를 가지고 있지 않은 현재의 방법들로는 결과에 대한 성능 보장을 할 수 없다는 것을 보았다. 이를 해결하기 위해 필요한 일은 우선 MDP가 얼마나 풀기에 효율적인지를 측정할

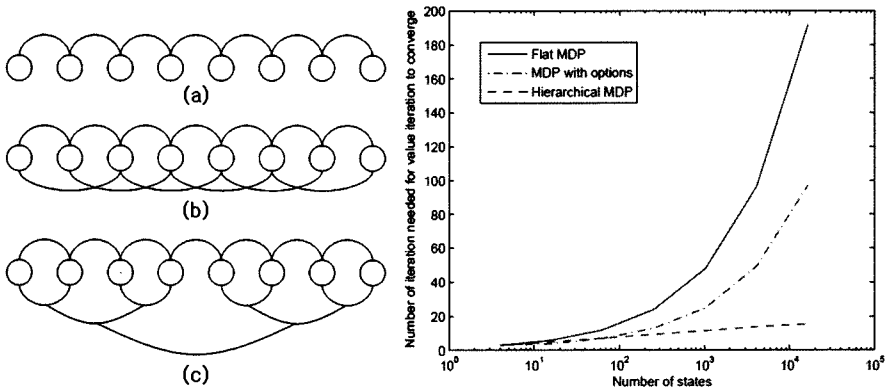


그림 1 세 종류의 MDP의 구조의 예와 MDP의 구조에 따른 사이즈와 풀이 시간의 관계. (a) 일반 MDP (b) 선택권이 추가된 MDP (c) 계층 구조가 추가된 MDP

수 있는 방법을 찾아내고, 이를 바탕으로 MDP를 보다 효율적으로 풀도록 할 수 있는 시간적 추상화 방법을 만드는 것이다.

본 연구에서 우리의 가설은 MDP의 위상학적 성질이 그 풀이 성능과 밀접한 관계가 있다는 것이다. 이를 입증하기 위해서는 MDP의 위상학적 성질을 측정해야 하는데, 이는 MDP로부터 아래에 설명한 바와 같이 상태 궤적 그래프를 얻어낸 후, 그 그래프에 그래프 이론적인 방법을 사용하여 측정하는 방법으로 구현이 가능하다.

상태 궤적 그래프는 각 노드(Node)가 상태 s, 각 변(Edge)이 행동 a에 대응되는 방향성 그래프이며, 이는 MDP의 시간에 따른 궤적(Trajectory)으로부터 얻을 수 있다. MDP의 궤적은 MDP의 상태 변화를 나타내는 순서쌍 $(s_i, a_{i,k}, s_j)$들의 집합으로 정의되고 여기서 s_i는 초기 상태, $a_{i,k}$는 그 상태에서 선택 가능한 행동들 중 행해진 행동, s_j는 그 결과로 이동한 상태를 나타낸다. 새로운 각 순서쌍 $(s_i, a_{i,k}, s_j)$에 대하여 상태 궤적 그래프에 노드 s_i와 s_j가 새롭게 추가되게 되고, 두 노드 간의 변 $a_{i,k}$가 생성되게 된다. 만일 상태 변화가 결정론적인(Deterministic) 단순화된 모델의 경우 변 $a_{i,k}$의 가중치(Weight) $w_{i,k}$는 1이 되고 각 변은 상태-행동 쌍 (s,a)에 일대일 대응되며, 상태 변화가 확률적인 경우 각 변의 가중치(Weight) 는 통계적으로 구할 수 있는 상태 변화 확률 $p(s_j|s_i,a_{i,k})$로 설정할 수 있다.

2.3 네트워크 척도와 MDP의 풀이 효율 간의 관계

실세계의 여러 네트워크들은 다양한 위상학적 특징을 가지고, 이 위상학적 특징들에 의해 네트워크의 연결정도와 네트워크에서 전달되는 정보의 동영학적인 성질이 영향을 받게 된다. 이들 위상학적 성질을 나타내어 네트워크의 분석, 구별, 생성 등에 쓰일 수 있는 것이 네트워크 척도이며, 네트워크의 종류와 원하는 분석 목표에 따라 다양한 종류의 네트워크 척도를 골라 사용할 수 있다[14].

앞서 세운 MDP의 위상학적 성질이 그 풀이 효율과 밀접한 관계가 있다는 가설을 입증하기 위해, 다양한 구조와 크기를 갖는 MDP를 생성한 뒤 그 상태 궤적 그래프에 다음과 같은 여러 네트워크 척도들을 사용하여 MDP들의 위상적인 성질을 수치화하고 각 척도들과 MDP들의 풀이 시간간의 상관관계를 분석하였다. 사용된 MDP와 네트워크 척도들의 종류, 그리고 MDP풀이에 사용된 알고리즘은 표 1과 같다.

정사각형 모양의 연속된 2차원 공간 [0,1) × [0,1)을 위의 네 가지 모델들을 사용하여 그래프 형태로 이산화하였다. 이산화시에 상태의 수는 4개부터 2500개까지, 모델의 분기 지수(Branching parameter)는 0.1로부터 0.9까지를 사용하여 다양한 크기와 종류의 MDP들을 만들었다. 전체 공간의 1%에 해당하는 [0,0.1) × [0,0.1)에서 보상 1이 주어졌고 나머지에선 보상 0이 주어졌으

표 1 실험에 사용된 네트워크 모델, 척도 및 MDP 풀이 알고리즘

사용된 네트워크 모델	▪ Regular lattice network ▪ Erdos-renyl network[15] ▪ Watts-Strogatz network[16] ▪ Barabasi-Albert network[17]
사용된 네트워크 척도	▪ 평균 차수(average degree) ▪ 최대 차수(maximum degree) ▪ 평균 측지 거리(mean geodesic distance) ▪ 전역 효율(global efficiency) ▪ 클러스터링 계수(clustering coefficient) ▪ 부분그래프 중앙도(subgraph centrality) ▪ 프랙탈 차원(fractal dimension)
사용된 MDP 풀이 알고리즘	▪ 평가치 반복 (Value iteration) ▪ Q-학습 (Q-Learning)

표 2 측정된 네트워크 척도들과 알고리즘들의 풀이 시간간의 상관관계

사용된 네트워크 척도	평가치 반복	Q-학습
평균 차수	-0.3413	-0.1467
최대 차수	-0.4607	-0.2426
평균 측지 거리	0.9913	0.9404
전역 효율	-0.6389	-0.6264
클러스터링 계수	0.3237	-0.3292
부분그래프 중앙도	0.4309	-0.2620
프랙탈 차원	-0.0398	0.1577

며, $\gamma=0.9, \epsilon=0.1$의 파라미터를 사용하여 각 알고리즘들이 최적치의 90%까지 수렴하는데 걸리는 반복 회수를 측정하였다. 다양한 크기와 종류의 MDP들을 사용하여 얻어진 MDP의 풀이 시간과 MDP로부터 구한 여러 네트워크 척도들은 각각 산포도(Scatter plot) 형태로 시각화하였고, 이들 간의 상관관계를 수치화하기 위해 양자 간의 피어슨 계수(Pearson coefficient)를 측정하였다. 그 결과는 표 2와 같다.

표 2에서 볼 수 있듯이, 테스트한 여러 네트워크 척도들 중 평균 측지 거리가 두 알고리즘 공통적으로 풀이 시간과 매우 높은 상관관계를 보였다.

2.4 이론적 분석

위에서 네트워크 척도 중 하나인 평균 측지 거리가 MDP의 일반적인 풀이 성능과 높은 상관관계를 보인다는 사실을 실험적으로 볼 수 있었는데, 이 사실을 뒷받침하기 위해 MDP의 풀이 성능과 평균 측지 거리와의 관계를 이론적으로 분석해 보았다. 분석을 위하여 가장 분석하기 쉬운 상태 s에서 행동 a를 취할 경우 고정된 보상 r과 고정된 다음 상태 s'를 얻는 결정론적 MDP, 그리고 그 중에서도 양의 보상이 한 상태에서만 주어지고, 나머지 상태에서는 보상이 0인 단일 보상 MDP (Single-goal MDP)에 대해 먼저 분석해 보고, 점차 일반적인 모델에 대해 확장해 나가도록 한다.

2.4.1 결정론적, 단일 보상 MDP(Deterministic single-reward MDP)

평가치 반복의 경우, 식 (1)의 갱신 규칙에 의하여 각 상태는 매 단계마다 주위의 모든 상태들에 대해 백업(backup)을 하게 된다. 따라서 특정 상태 s 의 평가 함수 $V(s)$는 보상이 주어지는 상태 s_{goal}로부터의 측지 거리 $GD(s,s_{goal})$만큼의 단계가 지난 후에 갱신되게 되며, 양의 보상이 한 상태에서만 주어지기 때문에 이 $V(s)$값은 더 이상 변경되지 않는다. 따라서 $V(s)$값은 $GD(s,s_{goal})$ 만큼의 시간 만에 구해지게 되며, 평균적인 풀이 시간은

$$\frac{1}{|s|}\frac{1}{|s|}\sum_{s_1}\sum_{s_2} GD(s_1,s_2) = MGD \qquad (5)$$

즉 단일보상의 결정론적 MDP의 경우 평가치 반복 알고리즘의 기대 수행 시간은 MDP의 상태 변화 그래프의 평균 측지 거리와 일치하게 된다.

Q-학습의 경우 식 (2)의 갱신 규칙에 의해 정책 외 백업(Off-policy backup)을 행한다[4]. 따라서 행동을 선택하는 정책에 따라 결과가 달라질 수 있다. 대표적으로 사용하는 정책인 ϵ-greedy의 경우, 확률 ϵ에 따라 임의 탐색(Random search)을 행하고, 나머지의 경우 현재까지 발견된 최적인 행동을 취하는 정책이다. 이 경우, 상태 경로 그래프 상의 최대 차수를 $d_{\max}$라고 하면 각 상태 s가 주위 모든 상태를 백업하는데 걸리는 기대 시간은 $\frac{d_{\max}}{\epsilon}$의 상한(Upper bound)을 갖게 된다. 이 경우 특정 상태 s의 평가 함수 $V(s)$가 구해지는 기대 시간은 $\frac{d_{\max}}{\epsilon}GD(s,s_{goal})$의 상한을 가지게 되며, 총 풀이 시간의 기댓값의 상한은 다음과 같이 얻어진다.

$$\frac{d_{\max}}{\epsilon}\frac{1}{|s|}\frac{1}{|s|}\sum_{s_1}\sum_{s_2} GD(s_1,s_2) = \frac{d_{\max}}{\epsilon}MGD \qquad (6)$$

정리하면 결정론적 단일 보상 MDP의 경우, 평가치 반복의 풀이 시간의 상한이 상태 변화 그래프의 평균 측지 거리에 선형적으로 의존(Linearly dependent)하며, Q-학습의 경우에도 작은 $d_{\max}$를 가지는 MDP의 경우 풀이 시간의 기대치가 MDP의 상태 변화 그래프의 평균 측지 거리에 유계(bounded) 된다는 것을 알 수 있다.

2.4.2 비 결정론적, 단일 보상 MDP(Stochastic single-reward MDP)

보다 일반적인 비 결정론적 MDP(stochastic MDP)는 상태 s에서 행동 a를 취할 경우의 결과가 고정되어 있지 않고, 상태 변화 함수 $T(s,a,s')$와 보상 함수 $r(s,a,s')$에 의해 주어지는 MDP이다. 이러한 경우의 한 극단적인 예로는 $T(\bullet,\bullet,s)$의 모든 수치가 0보다 큰 경우가 되겠는데, 이 경우에는 상태 변화 그래프가 완전 그래프(Complete graph)가 되어 MDP의 위상 구조를 따지는 의미가 사라지게 된다. 따라서 본 연구에서는 보다 제한적인 경우인, 상태 s의 다음 상태 s'의 경우의 수가 문제의 크기보다 작은 상한 $d_{\max}$를 갖는 비 결정론적 희소(Sparse) MDP에 대해서만 논의를 전개하도록 한다.

이 경우 $T(\bullet,\bullet,s)$들 중 0이 아닌 최소값을 $t_{\min}$이라 하면, 상태 경로 그래프 상에서 최대 차수는 $d_{\max}$가 되고, 특정 상태 s로부터 가능한 모든 주위 상태들로 백업하는데 걸리는 기대 시간의 상한은 위와 같은 과정을 통하면 평가치 반복의 경우 $\frac{1}{t_{\min}}$, Q-학습의 경우

$\frac{d_{\max}}{\epsilon t_{\min}}$이 되게 된다. 따라서 비 결정론적 회소 MDP의 경우 평가치 반복과 Q-학습의 수행 기대시간의 상한은 각각 다음과 같다.

$$\frac{1}{t_{\min}}\frac{1}{|s|}\frac{1}{|s|}\sum_{s_1}\sum_{s_2}GD(s_1,s_2) = \frac{MGD}{t_{\min}} \quad (7)$$

$$\frac{d_{\max}}{\epsilon t_{\min}}\frac{1}{|s|}\frac{1}{|s|}\sum_{s_1}\sum_{s_2}GD(s_1,s_2) = \frac{d_{\max}}{\epsilon t_{\min}}MGD \quad (8)$$

즉 비 결정론적 MDP의 경우에도 특정 조건 하에서는 풀이 시간의 기대치가 역시 MDP의 상태 변화 그래프의 평균 측지 거리에 유계된다는 것을 알 수 있다.

2.4.3 다중 보상 MDP(Multiple-reward MDP)

앞서 다룬 단일 보상 MDP보다 일반적인 형태는 양의 보상이 여러 군데에서 주어질 수 있는 다중 보상 MDP(Multi-reward MDP)이다. 이 경우의 문제는 상태 경로 그래프 상에 보상이 0이 아닌 Loop가 존재할 경우 상태 함수가 유한한 단계 안에 수렴하지 않을 수 있다는 것이다.

앞서 전개한 바와 같이 보상이 희소한 경우, 즉 보상이 0 이상인 Loop가 존재하지 않고 보상이 주어지는 상태들 $s_{r_1},s_{r_2},\ldots s_{r_{\max}}$의 수가 총 상태의 수보다 작은 상한 $r_{\max}$를 갖는 경우를 가정하면, 특정 상태 s의 평가함수 $V(s)$는 결정론적 MDP의 경우 $\max_i GD(s,s_{r_i})$의 시간 만에 구해지게 되고, 이는 다시 상태 변화 그래프의 최대 측지 거리, 혹은 반경(radius)에 유계되게 된다. Q-학습의 경우와 위에서 가정한 조건들을 만족하는 비 결정론적 회소 MDP의 경우에도 비슷한 방법으로 풀이 시간의 기대치가 $\max_i GD(s,s_{r_i})$에 비례하는 상한을 가짐을 보일 수 있다. 즉 다중 보상 MDP의 경우 단일 보상 MDP와 다르게 평균 측지 거리가 아닌 최대 측지 거리에 의해 수행 시간이 유계되게 된다.

2.4.4 일반적인 MDP

위에서 살펴본 바와 같이 보상이 단일 상태에서만 이루어지는 단일 보상 MDP의 경우 특정 가정 하에서, 평가치 반복과 Q-학습 두 풀이 알고리즘의 풀이 시간이 MDP의 상태 궤적 그래프의 평균 측지 거리의 상수 배에 의해 유계된다는 사실을 알 수 있었다. 대부분의 응용문제들에서 상태들은 소수의 인접 상태들과만 연결되어 있는 경우가 많고, 금기 검색(Tabu search) 등 탐색(Exploration)을 돕는 여러 방법들을 사용할 경우 상태 백업이 빠르게 행해지기 때문에, 위에서 세운 가정들은 상당히 현실적이라고 할 수 있고 실제 문제에서도 평균 측지 거리와 알고리즘의 풀이 시간이 매우 높은 상관관계를 보이리라 예측할 수 있다. 위의 예외 사항인 다중 보상 MDP의 경우에도, 평균 측지 거리가 낮은 그래프들의 경우 최대 측지 거리 또한 낮으리라 생각할 수 있으므로, 위 가정이 엄격히 성립하지 않는 실제 일반적인 MDP에 대해서도 평균 측지 거리와 MDP의 풀이 성능은 높은 상관 관계를 보이리라 추정할 수 있다.

3. 작은 평균 측지 거리를 갖는 상태 표현 모델

3.1 그래프 기반 상태 표현 모델

연속적인 공간에 강화 학습을 적용하기 위해서는 상태 공간을 이산화하거나 함수 근사장치를 사용하는 방식이 일반적이다. ITPM(Incremental Topology Preserving Map)은 이러한 함수 근사장치의 일종으로, 알려진 상태 공간을 일정한 영역을 가지는 노드(Node)들의 영역으로 나눈다[18]. 노드들의 위치는 자기조직화를 사용해 재배치되게 되고, 각 공간간의 연결 관계는 노드들 간의 변으로 표시된다. 이 ITPM의 경우 공간을 보로노이 다이어그램(Voronoi diagram)으로 분할하고, 분할된 공간간의 연결 관계도 얻을 수 있어 얻어진 노드들과 에지로 이루어진 그래프 상에서 위상적 경로탐색(Topological navigation)이 가능하기 때문에 경로 탐색 문제에 적합한 특징을 갖는다. 하지만, 복잡한 환경을 나타내기 위해서는 많은 노드들이 요구되는데 이 경우 근사장치를 사용하더라도 학습 시간이 문제 크기에 따라 증가하게 된다. ITPM의 구현 방법은 아래 표 3, 4에 제시되어 있다.

표 3 ITPM 알고리즘

1. 행동 a를 행하고 다음 상태 s'와 보상 r을 받는다.
2. ITPM에서 s'에 가장 가까운 노드 n'를 찾는다.
3. s'가 n'에서 ITPM의 반경 r 이상 떨어져 있을 경우 새로운 노드를 그 위치에 생성하고 5번으로 간다.
4. n'의 Q값을 사용해서 다음의 행동 a'를 선택한다.
5. RL 알고리즘을 사용해서 기존의 가장 가깝던 노드 n의 Q값을 수정한다.
6. 자기조직화: n'의 연결 상태와 위치를 수정한다.

표 4 ITPM의 자기조직화 알고리즘

1. 새로운 노드 n이 추가되었을 경우
   (a) n과 n',u와 n''를 연결하는 변을 만든다.
   (b) n'과 n'' 간의 변을 제거한다.
   추가되지 않았을 경우
   (c) n'과 n''를 연결한다.
2. 노드 n'과 n''에 인접해 있는 노드들 m을 s'쪽으로 이동시킨다.
   $w_{n'} <- w_{n'} + \delta(s' - w_{n'})$
   $w_m <- w_m + \delta_r(s' - w_m)$

3.2 복잡계망 모델(Complex network model)

대부분의 실세계 네트워크들은 작은 세상 성질(Small

world property), 높은 클러스터 계수(High cluster coefficient), 척도 없는 도수 분포(Scale-free degree distribution)와 같은 세 가지 성질을 가지고 있기 때문에, 이러한 성질들을 가지는 네트워크를 모델링하려는 많은 시도가 있어 왔다. 처음 제안된 방법인 [16]에서는 각 노드들이 이웃하고만 연결되어 있는 바둑판 모양의 격자에서 출발해서 임의의 확률로 링크를 추가한다. 생성되는 모델은 작은 세상 성질과 높은 클러스터 계수를 가지게 된다. 여기 추가하여 척도 없는 도수 분포를 가진 작은 세상 네트워크를 모델링하기 위해 [17]에서는 부익부 빈익빈 모델을 사용한 생장 네트워크 모델을 제안한 바 있다.

또한 사람의 경우 개인이 전체 네트워크에 대해 모르더라도 어느 경로가 더 가능성이 있는지 판단함으로써 짧은 경로를 찾아낼 수 있다는 사실에 기인해서 복잡계망에서 효율적인 비 중앙집중적(Decentralized) 탐색 알고리즘이 연구되어 왔다. [16]의 모델에서는 이러한 짧은 경로를 빠른 시간에 찾아내는 비 중앙집중적 알고리즘이 존재할 수 없다는 것이 증명되어 있으나, 이 모델의 내부 구조를 탐색에 사용할 수 있도록 수정하면 그러한 알고리즘이 가능하다[19]. 또한 내부 구조를 탐색에 전혀 사용할 수 없을 경우라도 네트워크 척도 없는 도수 분포를 가진다면 효율적인 탐색이 가능하다는 것도 알려져 있다[20].

3.3 복잡계망 모델을 이용한 그래프 기반 상태 표현 모델

앞서 말한 바와 같이 함수 근사장치를 사용할 경우에도, 상태 공간이 복잡해질 경우 학습해야 하는 함수 근사장치의 파라미터 수가 지수적으로 늘어나 학습하는 데 걸리는 시간이 크게 늘어나게 됨이 알려져 있다. 본 논문에서는 이에 대처하기 위해 고안된 작은 세상 성질을 가진 자기 조직화 성장 신경망 모델인 SW-ITPM을 사용한다. 작은 세상 성질이란 네트워크의 사이즈가 커지더라도 평균 측지 거리가 크게 증가하지 않는(대수 다항식적 비례 축소를 보이는) 성질을 의미하며, 앞서 보인 바와 같이 MDP의 풀이 성능은 평균 측지 거리와 높은 상관관계를 보이기 때문에, 이러한 작은 세상 성질을 가지는 복잡계망 모델을 이용할 경우 MDP의 우수한 풀이 성능을 얻을 수 있으리라 기대할 수 있다.

SW-ITPM은 ITPM의 자기 조직화 부분을 아래 표 5와 같이 수정한 것이다. 여기서 Model 1은 Kleinberg 모델[20], Model 2는 Barabasi-Albert 모델[17]에 대응된다. Kleinberg 모델은 균일한 격자 형태의 그래프에 거리에 따라 다른 확률로 변을 추가하는 방식이고, Barabasi-Albert 모델은 그래프를 생성해 나가면서 각 노드의 차수에 따라 다른 확률로 변을 추가하는 방식으로, 균일한 격자 형태의 그래프를 단계적으로 형성해 나가는 ITPM 알고리즘에 자연스럽게 접목이 가능하다.

표 5 SW-ITPM의 자기조직화 알고리즘

1. 새로운 노드 n이 추가되었을 경우
   (a) n과 n',u와 n''를 연결하는 변을 만든다.
   (b) n'과 n'' 간의 변을 제거한다.
   (c) 노드 o를 다음에 비례하는 확률분포에 따라 선택한다.
   MODEL 1: $distance(n,o)^{-p}$
   MODEL 2: $\mathrm{degree}(o)$
   (d) 확률 p_{ll}로 n과 o를 연결한다.
   추가되지 않았을 경우
   (e) n'과 n''를 연결한다.
2. 노드 n'과 n''에 인접해 있는 노드들 m을 s'쪽으로 이동시킨다.
   $w_{n'} \leftarrow w_{n'} + \delta(s' - w_{n'})$
   $w_m \leftarrow w_m + \delta_r(s' - w_m)$

3.4 그래프 기반 상태 표현 모델을 사용한 강화 학습

Q-학습과 같은 대부분의 강화 학습 알고리즘의 경우 행동 선택 방법에는 큰 제약이 주어지지 않는다. 실제로 행동 선택 방법과 상관없이 모든 상태가 충분한 회수 이상 방문될 경우 최적의 해를 구할 수 있다. 그래서 일반적으로 일정 확률로 무작위 행동을 하고 나머지의 경우 알려진 것 중 가장 좋은 것을 찾는 ϵ-greedy 방식 등이 주로 쓰이고 있다. 하지만 네트워크의 구조에 따라 적절한 비 중앙집중적 탐색 알고리즘을 사용하여 탐색의 효율을 한층 더 높일 수도 있다. Model 1의 경우 노드의 거리를 행동 선택에 이용할 경우 효율적인 탐색이 가능함이 알려져 있다. 반면 척도 없는 도수 분포를 가지는 Model 2의 경우 무작위로 이동하더라도 더 높은 차수로 확률적으로 이동하게 되고, 따라서 별도의 탐색 알고리즘을 사용하지 않아도 효율적인 탐색이 가능해짐이 알려져 있다. 또한, 원래의 네트워크에 링크가 추가된 네트워크는 시간적 추상화가 적용된 MDP와 대응되므로, 세미-MDP에 사용되는 평가 함수 갱신 알고리즘들 (3), (4)를 그대로 사용할 수 있다.

4. 실험 및 결과

4.1 2D Puddleworld 문제

제안된 알고리즘의 동작을 테스트하기 위하여, 우선 강화 학습에서 테스트 문제로 많이 쓰이는 2차원 탐색 문제인 Puddleworld 문제를 사용하였다. 이 문제는 T자 모양의 장애물이 설치된 연속된 2차원 공간을 사용하며, 학습 에이전트는 임의의 상태에서 출발하여 보상을 최대화하는 경로를 찾게 된다.

4.1.1 실험 세팅

[0,1) × [0,1)의 2차원 상태 공간을 사용하였고, 보상

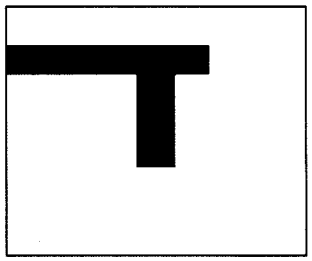


그림 2 2D Puddleworld 문제

r은 [0,0.1) × [0,0.1)의 목표지점에서만 10, 나머지의 경우 -1이 주어졌다. ITPM과 모델 1,2를 사용한 두 가지 SW-ITPM을 사용하여 그래프 형태로 환경을 학습하고 Q-학습 알고리즘을 사용하여 이를 풀었다. ITPM에는 $r=0.05/0.025/0.0125/0.00625$, $\delta=0.0002$, $\delta_r=0.00002$, $p=2.322$, SW-ITPM에는 $p_{ll}=0.2$, 그리고 Q-학습에는 $\epsilon=0.1$, $\alpha=0.5, \gamma=0.7$의 파라미터가 사용되었다.

4.1.2 실험 결과

학습된 그래프의 모양과 도수 분포는 그림 3과 같다. 3(a)로부터 ITPM은 평균 차수가 5에 집중된 균일한 격자 모양의 네트워크를 형성함을 알 수 있다. 모델 1, 모델 2를 적용한 경우 생성되는 추가 링크를 나타낸 3(b), 3(c)의 결과를 보면 각 모델의 특성대로 모델 1은 노드들의 차수가 낮지만, 모델 2의 경우 높은 차수의 허브 노드들이 생성됨을 볼 수 있다.

문제 사이즈에 따른 노드들 간의 평균 거리는 그림 4와 같다. 복잡계망 모델을 사용하지 않은 경우 노드간 거리가 문제 사이즈에 비례해서 증가하나 모델 1, 모델 2를 사용한 경우 노드 간 거리가 문제 사이즈에 지수적으로 비례해서 증가하는 작은 세상 성질을 보임을 알 수 있다.

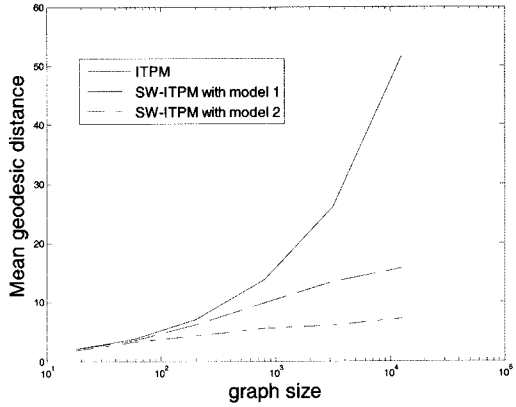
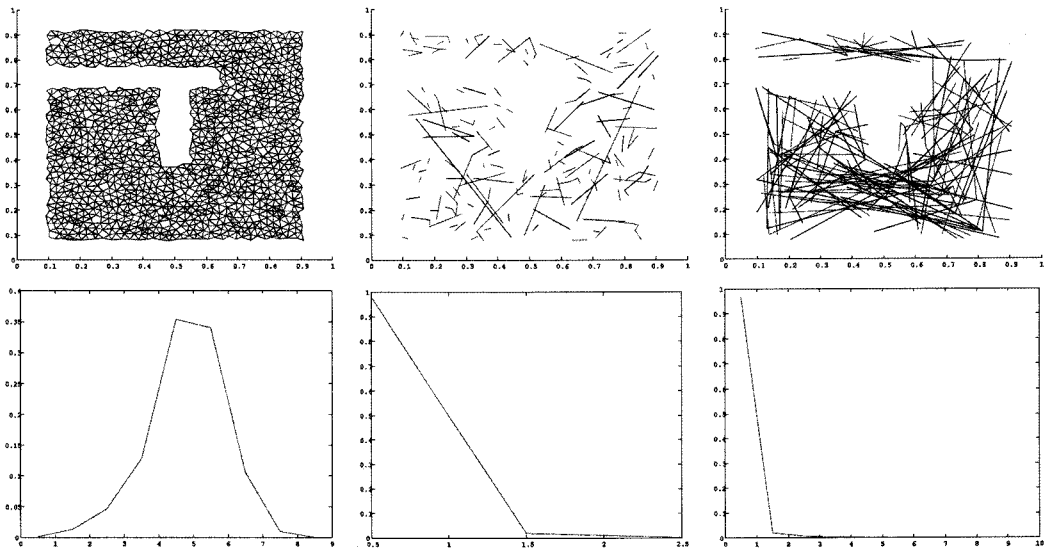


그림 4 문제 크기에 따른 모델별 평균 측지 거리

4.1.3 학습 성능 비교

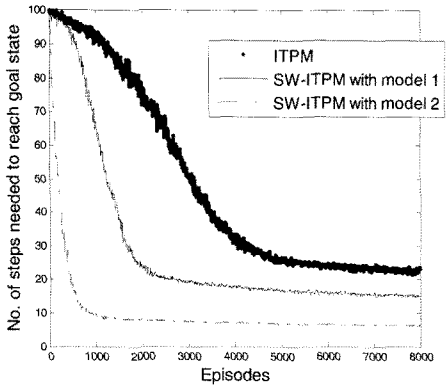
학습된 세 모델을 사용하여 Q-학습 알고리즘을 수행하여 학습 성능을 비교하여 보았다. 그림 5에 문제 사이즈에 따른 강화 학습의 학습 곡선이 나와 있다. 사이즈가 커짐에 따라 복잡계망 모델을 사용할 경우 수렴속도가 크게 빨라짐을 알 수 있다.

문제 사이즈에 따른 학습 시간을 비교하기 위해, 최종 수렴 값의 절반에 다다르기까지의 반복 횟수를 평균 학습 시간으로 정의하고 사이즈에 따라 비교하였다(그림 6).

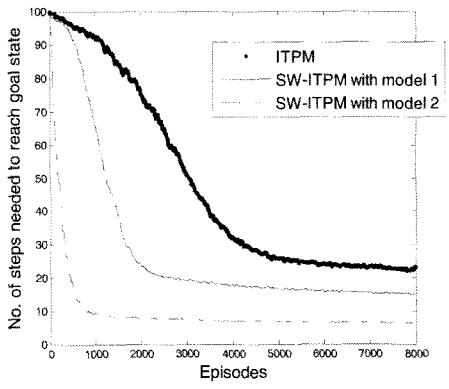


(a) (b) (c)

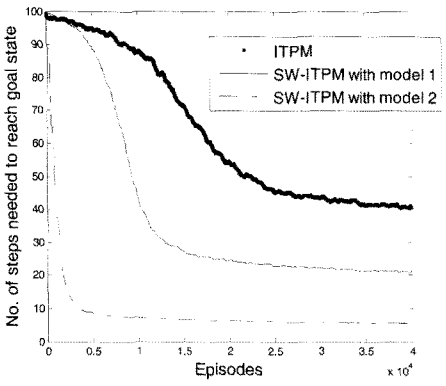
그림 3 각 모델별로 생성된 그래프와 그 도수 분포. (a) ITPM (b) SW-ITPM, Model 1 (c) SW-ITPM, Model 2



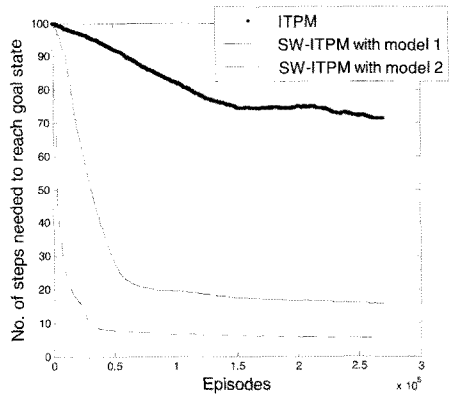
(a) r=0.05



(b) r=0.025



(c) r=0.0125



(d) r=0.00625

그림 5 문제 사이즈에 따른 모델별 학습 곡선

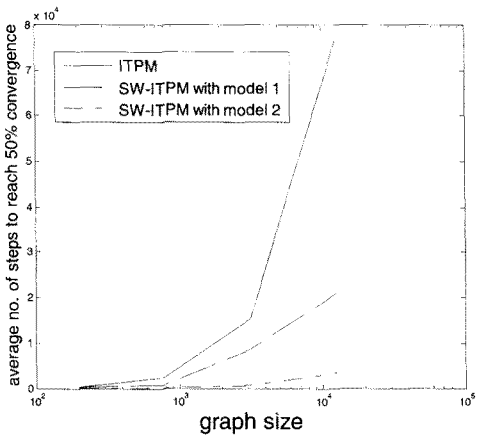


그림 6 문제 사이즈에 따른 모델별 학습 시간

복잡계망 모델을 사용한 경우 사용하지 않은 경우보다 학습 시간의 증가폭이 현저하게 줄어들었음을 알 수 있다. 모델 1과 모델 2를 비교해 보면 문제 사이즈에 따른 노드 간 평균 거리의 변화의 양상은 거의 유사하지만 강화 학습의 성능은 모델 1이 크게 떨어지는데, 이는 모델 2가 척도 없는 도수 분포를 가짐으로써 별도의 탐색 알고리즘 없이 효율적인 탐색이 가능하기 때문이라고 생각된다.

4.2 3D Gameworld 문제

보다 사이즈가 크고 실제 환경에 가까운 문제에서 제안된 알고리즘을 테스트하기 위해, 연속적인 3차원 환경을 제공하는 게임인 Half-Life 2를 사용하여 실험하였다. 사용된 환경은 2층 실내 환경을 높은 정밀도로 묘사한 50*150*10m 사이즈에 대응되는 dm_lockdown 맵이고, 여기서 크기가 1m*1m*1m인 학습 에이전트가 환경 안에서 임의의 방향으로 움직이며 보상을 최대화하는 경로를 찾게 된다.

4.2.1 실험 세팅

에이전트는 최대 시속 7m/s로 연속으로 이동하고, 매 1/20초마다 현재의 위치와 주위 8방향의 1m이내의 장

애물 유무를 관측하여 평가치를 수정하고 다음의 진행 방향을 결정하게 하였다. 학습에는 평가치 반복 알고리즘을 사용하였고, 원활한 탐색을 위해 평가 함수는 낙관적 초기치(Optimistic initial value)를 사용하였다. 여러 대의 에이전트가 동시에 행동할 수 있으며, 이 경우 이들 에이전트는 ITPM을 공유하게 된다. ITPM에 사용된 파라미터는 $r=1m/0.75m/0.5m$, $\delta=0.0002$, $\delta_r=0.00002$, $p=\log_2 5$이고 평가치 반복에 사용된 파라미터는 $\epsilon=0.1$, $\alpha=0.5$, $\gamma=0.9$이다.

4.2.2 실험 결과 및 분석

우선 에이전트의 공간 탐색 능력을 테스트하기 위해 에이전트들을 맵의 여러 군데에 투입하였다. 각 에이전트들은 자기 주위의 공간부터 아직 가보지 않은 공간을 우선적으로 탐색해 나가며 ITPM을 만들어 나가게 된다. 생성된 ITPM의 예는 아래의 그림 7과 같다.

복잡계망 모델을 적용했을 경우와 그렇지 않을 경우의 정보 전파 속도를 비교하기 위해, ITPM과 두 SW-ITPM에 대해 평가치 반복 알고리즘의 매 단계별로 갱신

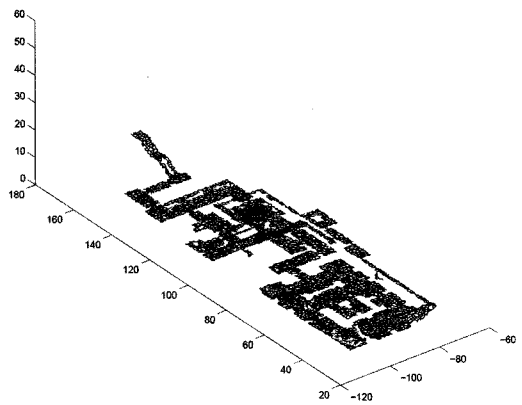


그림 7 3D 환경에 대해 학습된 ITPM

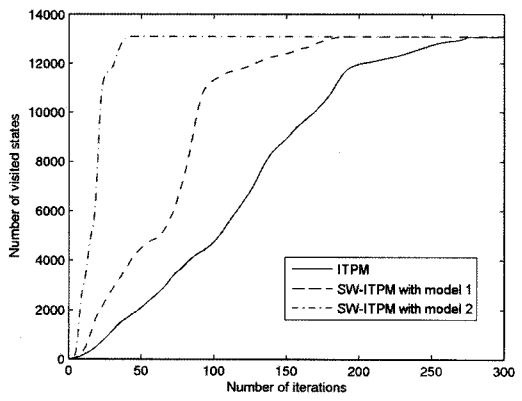


그림 8 각 모델의 정보 전파 속도의 비교

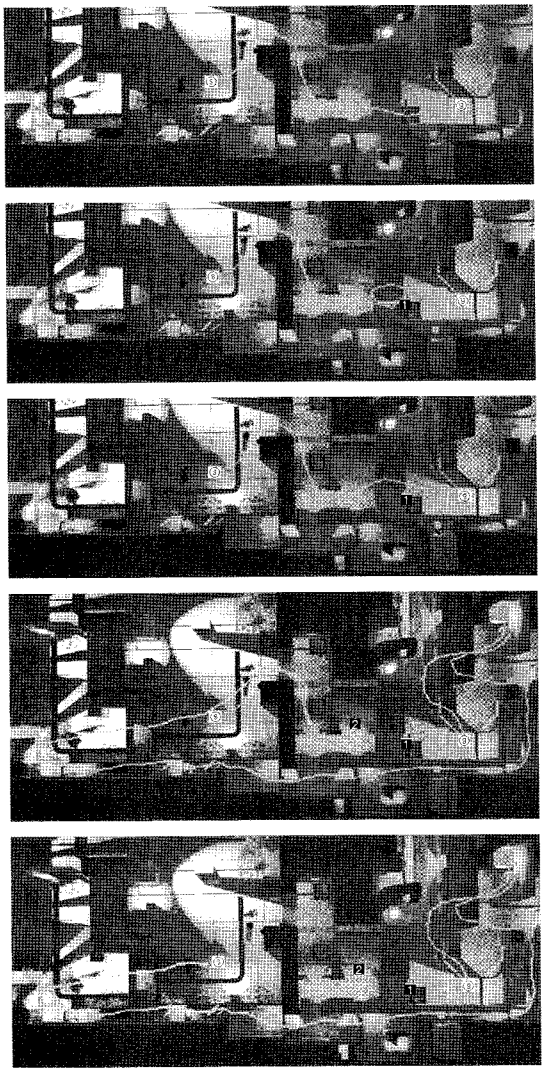


그림 9 장애물 1, 2를 설치 시 동적인 경로 재탐색 과정

된 상태의 수를 비교해 보았다(그림 8). 복잡계망 모델을 적용할 경우 정보의 전파 속도가 훨씬 빠름을 알 수 있다.

마지막으로 동적으로 변하는 환경에 대처하는 능력을 보기 위해, 우선 보상 1이 주어지는 목표 지점을 사용하여 최단 경로를 학습한 후, 경로에 장애물 1, 2를 차례로 설치하여 새로운 경로를 찾아내도록 하였다. 실험 결과 장애물의 존재를 인지한 에이전트는 실시간으로 새로운 최단 경로를 찾아내고, 그 후에는 더 이상의 시행착오 없이 새로운 경로를 사용하는 것을 알 수 있었다(그림 9).

각 모델들의 학습 성능을 비교하기 위해, 각각의 장애물에 대해 ITPM 모델과 두 SW-ITPM 모델들이 장애물을 처음 확인한 후 새로운 최단경로를 찾아내는 데 걸리는 시간을 측정하여 비교하였다. 또한 문제 사이즈

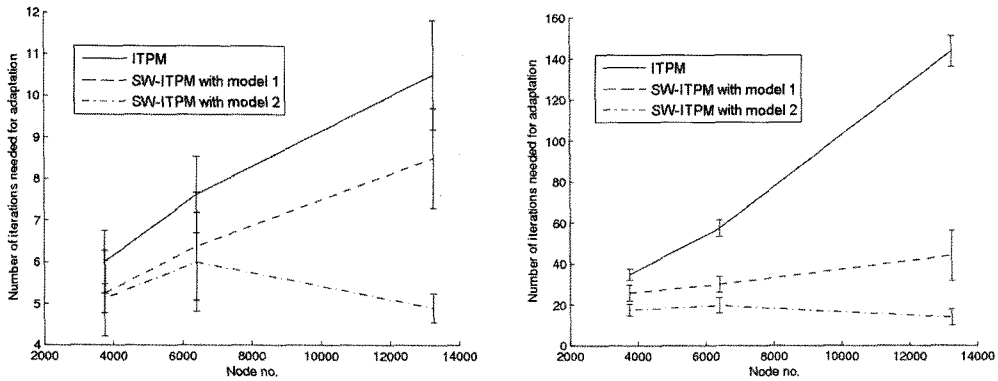


그림 10 장애물 1과 2에 대한 각 모델별 경로 재학습 시간

의 변화에 따른 학습 성능의 변화 양상을 보기 위해, 세 가지 ITPM 반경을 사용하여 크기가 다른 세 개의 모델을 사용하여 각각 성능을 테스트하였다.

실험 결과 그림 10에서 볼 수 있듯이 장애물 1과 같이 새로운 최단경로의 길이가 크지 않을 경우 SW-ITPM으로 인한 성능 개선이 크게 나지 않지만, 장애물 2와 같이 새로운 최단경로의 길이가 길 경우에는 SW-ITPM으로 인한 평균 측지 거리의 단축이 매우 큰 성능 향상을 가져오는 것을 볼 수 있다. 문제의 크기가 4배 증가하는 경우 ITPM만 사용할 경우 4배 이상의 시간이 소요되었지만, SW-ITPM의 경우 문제 크기의 변화에 따른 학습 시간의 변화가 훨씬 적었으며, 특히 모델 2의 경우 문제 크기에 학습 시간이 거의 영향을 받지 않는 모습을 보였다.

5. 결 론

본 논문에서는 기존 강화 학습 알고리즘들의 두 가지 큰 문제점인 연속적인 상태 공간과 차원성의 저주를 해결하기 위한 그래프 기반 상태 표현 모델을 제시하였다. 이를 위하여 우선 MDP의 일반적인 풀이 성능과 MDP의 위상적 성질간의 관계에 주목하고, MDP로부터 얻은 상태 궤적 그래프의 네트워크 척도들과 MDP의 일반적인 풀이 성능간의 관계를 분석하였으며, 그 결과 네트워크 척도들 중 하나인 평균 측지 거리가 MDP의 일반적인 풀이 성능과 높은 상관관계를 보인다는 것을 이론 및 실험적으로 보였다. 또한 이 사실을 기반으로 하여, 연속된 상태 공간을 표현할 수 있는 그래프 기반 상태 표현 모델과 낮은 평균 측지 거리를 갖는 복잡계망 모델을 결합하여 우수한 성능을 보장할 수 있는 그래프 기반 상태 표현 모델을 제시하였다.

실험 결과 제안된 알고리즘은 연속된 상태 공간으로부터 사용된 복잡계망 모델의 성질을 만족하는 이산화된 그래프 형태의 상태 표현 모델을 얻어낼 수 있음이 확인되었고, 이 모델을 사용하여 학습시 문제 사이즈가 커지더라도 기존과 같은 지수적인 성능 약화를 막을 수 있다는 것을 보였다.

차후 과제로는 각 모델에 적합한 보다 우수한 탐색 알고리즘을 사용함으로써 강화 학습의 성능을 보다 끌어올리고, 이를 실제 문제에 적용하는 것을 생각할 수 있다.

참 고 문 헌

[1] Barto, A. G., Mahadevan, S., "Recent advances in hierarchical reinforcement learning," *Discrete Event Systems Journal*, Vol.13, pp. 41-77, 2003.

[2] Sutton, R. S., Precup, D., Singh, S. P., "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning," *Artificial Intelligence*, Vol.112, pp. 181-211, 1999.

[3] Fritzke, B., "A growing neural gas network learns topologies," In Proc. of *the 7th Neural Information Processing Systems*, pp. 625-632, 1995.

[4] Sutton, R. S., "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, Vol.4, pp. 237-285, 1996.

[5] Watkins, C. J., Dayan, P., "Q-learning," *Machine Learning*, Vol.8, pp. 279-292, 1992.

[6] Littman, M. L., Dean, T. L., Kaelbling, L. P., "On the complexity of solving Markov decision problems," *Uncertainty in Artificial Intelligence*, pp. 394-402, 1995.

[7] Beleznay, F., Grobler, T., Szepesvari, C., "Comparing value-function estimation algorithms in undiscounted problems," 1999.

[8] Dietterich, T. G., "Hierarchical reinforcement learning with the MAXQ value function decomposition," *Journal of Artificial Intelligence Research*, Vol.13, pp. 227-303, 2000.

[9] Pickett, M., Barto, A. G., "Policyblocks: An algo-

rithm for creating useful macroactions in reinforcement learning," In Proc. of *the 9th International Conference on Machine Learning*, pp. 506-513, 2002.

[10] Digney, B., "Learning hierarchical control structure for multiple tasks and changing environments," In Proc. of *the 5th Conference on the Simulation of Adaptive Behavior*, 1998.

[11] McGovern, A., Barto, A. G., "Subgoal discovery for hierarchical reinforcement learning using learned policies," In Proc. of *the International Conference on Machine Learning*, pp. 361-368, 2001.

[12] Jong, N.K., Stone, P., "State abstraction discovery from irrelevant state variables," In proc. of *the 19th International Joint Conferences on Artificial Intelligence*, pp. 752-757, 2005.

[13] Simsek, O., Wolfe, A. P., Barto, A. G., "Identifying useful subgoals in reinforcement learning by local graph partitioning," In Proc. of *the 22nd International Conference on Machine Learning*, pp. 816-823, 2005.

[14] da F. Costa, L., Rodrigues, F. A., Travieso, G., Boas, P. R. V., "Characterization of complex networks: A survey of measurements," 2005.

[15] Erdos, P., Renyi, A., "On random graphs," *Publicationes Mathemticae (Debrecen)*, Vol.6, pp. 290-297, 1959.

[16] Watts, D. J., Strogatz, S. H., "Collective dynamics of 'small-world' networks," *Nature*, Vol.393, pp. 404-407, 1998.

[17] Barabasi, A.L., Albert, R., "Emergence of scaling in random networks," *Science*, Vol.286, pp. 509-512, 1999.

[18] Jose del R. Millan, Posenato, D., Dedieu, E., "Continuous-action q-learning," *Machine Learning*, Vol.49, pp. 241-265, 2002.

[19] Kleinberg, J., "The Small-World Phenomenon: An Algorithmic Perspective," In Proc. of *the 32nd ACM Symposium on Theory of Computing*, pp. 163-170, 2000.

[20] Adamic, L. A., Lukose, R. M., Puniyani, A. R., Huberman, B. A., "Search in power-law networks," *Phys. Rev. E*, Vol.64, pp. 46135-46143, 2001.

이 승 준

2002년~현재 서울대학교 전기·컴퓨터공학부 석·박사 통합과정. 관심분야는 강화학습, 지능형 에이전트, 제어이론, 기계학습, 가상현실

엄 재 홍

2001년 서울대학교 전기·컴퓨터공학부 석사. 2009년 서울대학교 전기·컴퓨터공학부 박사. 관심분야는 텍스트마이닝, 정보추출, 정보검색, 기계학습, 데이터베이스, 생물정보학, 의료정보학

장 병 탁

정보과학회논문지 : 소프트웨어 및 응용 제 36 권 제 3 호 참조