강화학습기법을 이용한 목적지 경로 탐색

이태경*, 전준리*
*동국대학교 컴퓨터과학과
e-mail : junri_@naver.com

Destination Path Search using Reinforcement Learning Technique

Taekyung Lee*, Junri Jeon*
*Dept. of Computer Science, Dongguk University

요 약

본 논문에서는 목적지들의 중요도를 이용하여 강화학습에 의한 목적지 경로 탐색을 제안한다. 일반적인 목적지 경로탐색은 목적지의 중요도나 방문빈도를 고려하지 않는 최단경로탐색을 수행한다. 그러므로 방문객들의 요구에는 맞지 않는 경로를 탐색한다. 강화학습의 특징은 관심 대상에 대한 구체적인 지배 규칙의 정보 없이도 최적화된 행동 방식을 학습시킬 수 있는 특징이 있다. 이를 이용하면 주요목적지를 누락시키지 않고 방문객들의 요구에 만족하는 경로를 탐색할 수 있다. 기존에 이용되고 있는 경로탐색 알고리즘과 강화학습기법이 적용된 알고리즘을 서로 분석하여 비교한다.

1. 서론

경로 탐색 알고리즘이 과거보다 현재에 더욱 주목받는 이유는 전자기기 및 휴대기기들의 발달로 우리들의 생활이 한층 더 가까워졌기 때문이다. 현대인들의 생활에서 주로 사용되는 경로 탐색 알고리즘은 자동차 내에서 사용하는 내비게이션에서 이용한다. 최근엔 이 한계를 벗어나 내비게이션이 스마트폰에서도 활용되고 있다. 내비게이션에 기본적으로 구현되는 경로탐색은 Dijkstra 알고리즘, A* 알고리즘, 유전자 알고리즘을 적용한다[1][2][3]. 그렇지만 목적지를 방문할 경우에는 최단경로 탐색만으로 방문객들을 만족하게 하기는 쉽지 않다.

이를 보완할 방법으로는 인공지능 기법을 사용하는 것이다. 인공지능의 계산적인 접근 방법을 바탕으로 인간이 할 수 있는 일을 기계도 할 수 있는 것이다. 인공지능 기법으로 A* 알고리즘, Heuristic 기법 등의 다양한 알고리즘들이 실생활에서 유용하게 이용되고 있고, 학문적으로 연구 되는 등 여러 분야에서 사용하고 있다[3][14]. ITS(Intelligent Transportation Systems) 서비스를 이용한 최적 교통 지원이 그 예라고 할 수 있다[4].

인공지능의 역할은 응용프로그램에서 에이전트를 이용하면 손쉽게 처리할 수 있다. 에이전트의 특성으로 자율성(autonomy), 사회성(social ability), 반응성(reactivity), 주도적 능동성(pro-activeness) 등을 가진다[5]. 특성의 성질들을 이용하면 사용자의 환경과 행동에 맞춰서 효율적으로 수행이 된다.

본 논문에서는 다양한 에이전트 기법인 강화학습기법을 이용한 경로탐색 알고리즘을 제안하고 기존에 많이 이용되고 있는 최단경로 알고리즘과 분석하여 비교한다.

2. 경로탐색과 문제해결

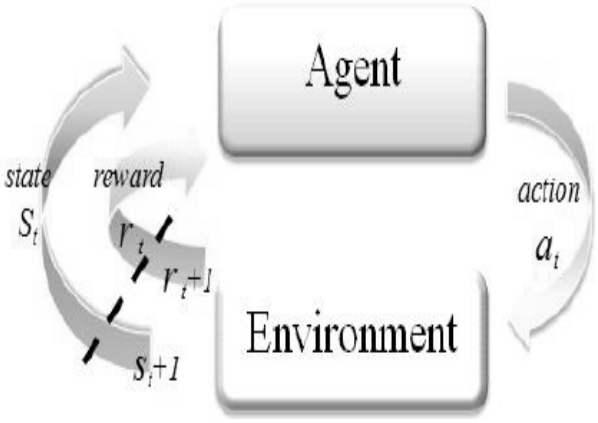
2.1 최단경로탐색 알고리즘

최단경로 알고리즘의 대표적인 알고리즘으로 Dijkstra 알고리즘이 있다. 이 알고리즘은 모든 간선의 가중치가 음이 아닐 때, 방향 그래프 $G=(V,E)$ 에서 단일 출발점 최단 경로를 해결하는 알고리즘이다. 방향 그래프의 V는 정점, E는 간선을 나타낸다. 가까운 정점부터 차례로 모든 정점에 대한 간선들의 합으로 최단 경로를 찾는다. 한 정점에서 시작하여 연결된 간선들의 가중치를 고려해서 하나의 간선을 선택하고, 이 간선에 연결된 정점을 추가하는 과정을 반복한다. 최단 경로를 찾는 방법은 탐욕적 전략을 사용하는데 가중치가 가장 가까운 정점을 선택하는 방법이다[6][10].

2.2 강화학습기법

강화학습이란 동적인 환경에서 에이전트가 시행착오(trial and error)를 거치면서 보상값을 받아 최적의 결과값을 찾아내는 학습이론이다[11].

(그림 1)의 형태로 학습자가 주어진 환경과 상호 작용을 할 때 상태(state), 행동(action), 보상(reward)이라는 세 가지 기본 요소를 이용한다[12]. 강화학습의 에이전트가 임의의 상태에서 행동을 실행할 때 가장 적절한 행동을 선택하고 보상을 한다.



(그림 1) 에이전트와 환경의 상호작용

2.3 Q-Learning

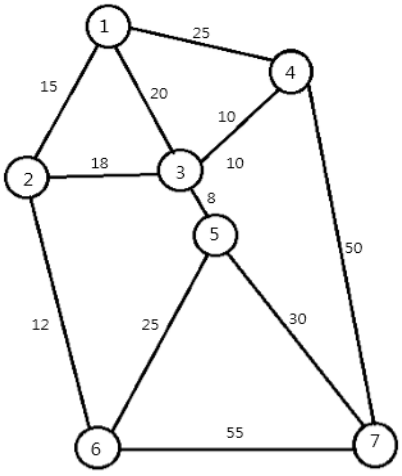
강화학습기법에서 널리 사용되는 방법의 하나로 Q-Learning이 있다. Q-Learning은 환경, 에이전트, 상태, 행동, 보상으로 구성된 강화 학습 알고리즘이다. Q-Learning은 강화학습 알고리즘의 하나로 행동가치함수(action-value function)를 이용해서 학습을 수행한다. 모든 행동을 수행해볼 필요 없이 환경상태에서 최적의 값을 구해낸다. 가치함수는 아래의 식으로 계산된다.

$$V(s) = \max_a Q(s,a) \quad (1)$$

위 식에서 s는 임의의 상태, a는 선택된 행동을 나타낸다. 임의의 상태 s에서의 가치함수는 그 상태에서 Q값이 최대로 되는 행동에 계산된 Q값으로 정의된다[7]. 학습에 의해 Q값을 갱신한다.

$$Q(s,a) = Q(s,a) + \alpha[R(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a)] \quad (2)$$

위 (2)의 식에서 R(s,a)은 상태 s에서 행동 a를 실행하였을 때 주어지는 보상값이고,. γ는 할인 상수(discount factor)로 미래에 받게 될 보상이 현재 상태의 가치나 상태-행동의 가치에 반영되는 정도를 조절한다[8]. α는 에이전트의 학습률이며, $0 \leq \alpha \leq 1$의 범위를 가지고 학습속도에 영향을 끼치는 파라미터이다. 만약 너무 작은 값을 가지면 학습이 천천히 진행되고, 너무 큰 값을 가지면 학습이 제대로 이루어지지 않게 된다[9]. 가능한 상태의 행동은 ϵ-greedy 로 결정한다. Q값 중 1-ϵ의 확률로 가장 큰 값을 선택하고 나머지는 ϵ확률로 무작위 행동을 취하게 된다($\epsilon \in [0,1]$)[13].

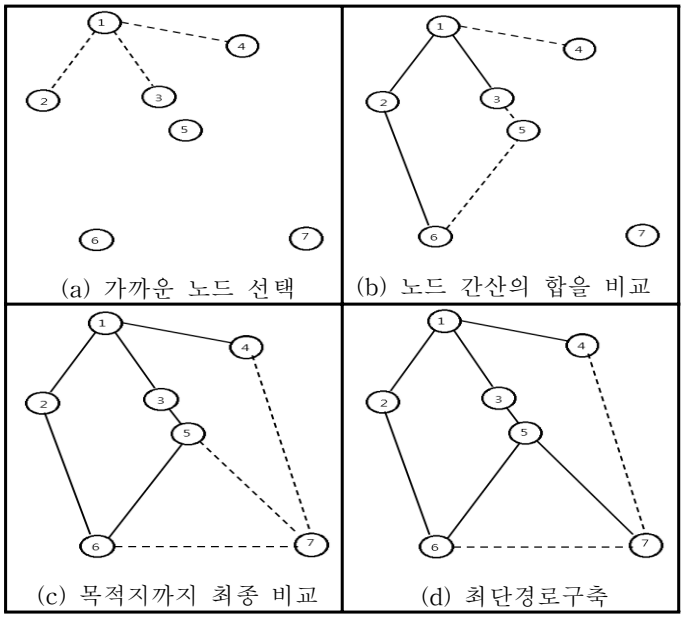


(그림 2) 모든 목적지 간의 거리

3. 알고리즘 분석 및 비교

3.1 Dijkstra 알고리즘

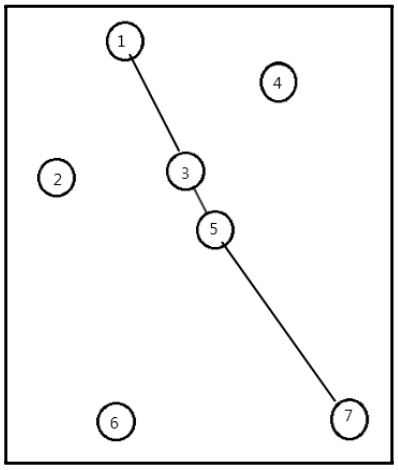
(그림 2)에서는 Dijkstra 알고리즘을 이용한 경로탐색과 강화학습을 적용한 경로탐색을 비교를 하기 위해 구성한 모든 목적지 간의 거리를 나타낸 그래프이다. (그림 3)에서는 출발지에서 목적지까지 가는 최단경로를 나타낸다. 다른 가중치는 고려하지 않고 출발지에서 목적지까지의 탐색을 Dijkstra 알고리즘을 적용하였을 때의 탐색과정은 다음과 같다.



(그림 3) 최단경로탐색 과정

(그림 3)과 같이 출발지 1번 노드에서 목적지 7번 노드로 이동하는 최단경로탐색이다. (그림 3)(a)는 간선의 최단거리를 비교하여 가장 짧은 정점을 선택하는 과정이다.

(그림 3)(b)는 간선을 추가하여 이전에 탐색한 최단경로를 비교하고, (그림 3)(c), (그림 3)(d)와 같이 반복적으로 탐색한다.



(그림 4) 최단경로탐색

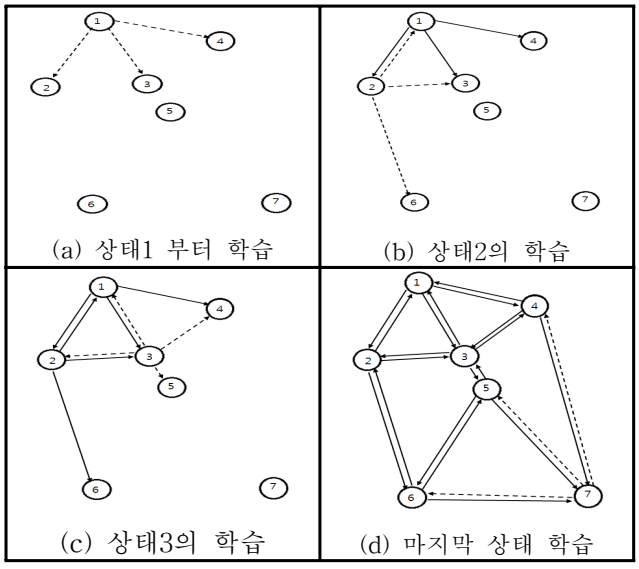
(그림 4)의 탐색결과를 통하여 출발지에서 목적지까지의 이동 거리는 아래의 <표 1>을 통해 확인할 수 있다.

<표 1> Dijkstra 알고리즘의 총 이동거리

(그림 3) 의 경로탐색 :	1→3→5→7
(그림 3) 의 이동거리 :	20+8+30=58

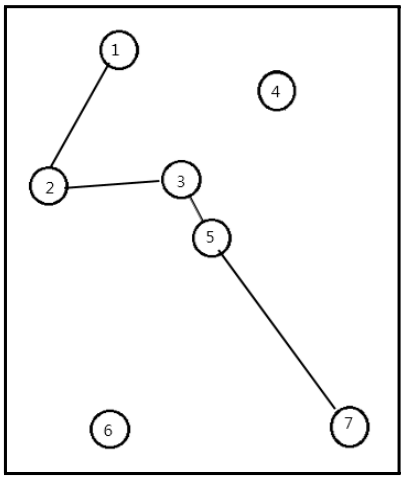
위 <표 1>에서 나타난 Dijkstra 알고리즘의 이동거리는 58이 된다. 1번 노드에서 7번 노드로 이동하는 가장 짧은 최단경로이다.

3.2 강화학습에 의한 최단경로 알고리즘



(그림 5) 강화학습이 진행되는 과정

(그림 5)는 강화학습이 진행되는 과정이다. (그림 5)와 같이 임의의 상태에서 Q-Learning을 실행한다. 1번 노드부터 7번 노드까지 모두 강화학습을 적용한다. 강화학습 적용 후 이동 경로는 다음과 같다.



(그림 6) 강화학습에 의한 최단경로탐색

(그림 6)에서는 강화학습을 적용한 후에 나온 최단경로 알고리즘이다. 출발지는 1번 노드이고, 목적지는 7번 노드이다. 각 노드의 가중치가 다르게 적용되었기 때문에 1번 노드에서 2번 노드를 방문한 후, 2번 노드에서 3번 노드로 방문하게 된다.

<표 2> 강화학습 후 목적지까지의 거리

(그림 4) 의 경로탐색 :	1→2→3→5→7
(그림 4) 의 이동거리 :	15+18+8+30=71

(그림 6)의 경로탐색은 2번 노드를 방문한 후 3번 노드를 방문하기 때문에 이동거리는 증가하였다. 하지만 2번 노드를 거쳐 가면서 방문객이 만족할 수 있는 만족도를 높일 수 있다.

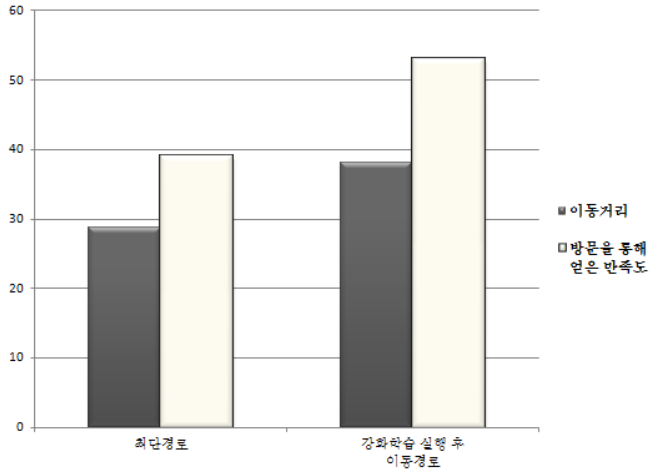
3.3 알고리즘 분석

(그림 6)에서는 최단경로탐색과 강화학습을 실행 후 이동거리와 방문을 통해 얻은 만족도를 나타낸다. 방문을 통해 얻는 만족도를 수치로 나타내기 위해 다음과 같은 가중치를 부여한다.

<표 3> 노드를 방문할 때 얻는 만족도

노드	1	2	3	4	5	6	7
가중치	14	17	18	14	14	4	9

위 <표 3>은 가중치로 노드를 방문하였을 때 얻는 이익이다. 각각의 노드에서 임의의 노드를 선택하였을 때 얻게 되는 평균 이동거리와 방문을 통해 얻는 만족도는 (그림 7)과 같다.



(그림 7) 두 알고리즘의 손익 비교

강화학습으로 경로를 구성하였을 때에는 최단경로를 탐색할 때보다 이동 거리가 늘어나는 손해가 발생하지만, 목적지를 방문하여 얻는 만족도가 더 높아지게 된다.

4. 결론 및 향후과제

본 논문에서는 일반적인 최단경로 알고리즘과 목적지 방문 시 얻는 만족도를 이용한 강화학습의 Q-Learning 알고리즘을 분석하고 비교하였다. 결과에서 보이는 것과 같이 Dijkstra 알고리즘을 사용했을 때보다 강화학습을 사용했을 때 이동거리가 증가하였다. 그렇지만 방문객들이 방문을 통해 얻는 만족도가 높아졌다. 비록 이동거리가 증가하는 단점이 있지만, 증가하는 이동거리에 비해 방문객이 방문을 통하여 얻는 만족도가 높아지게 된다. 목적지 경로탐색은 방문객의 만족도에서 Dijkstra 알고리즘보다 강화학습을 적용한 경로탐색이 더 우수하다.

현재 방문한 목적지로부터 다음에 방문할 목적지가 과다하게 먼 거리에 있을 경우에 전체적인 효율이 감소한다. 또한 방문지가 적을 때에는 다른 최단경로 알고리즘과 효율면에서 큰 차이가 없다는 단점이 있다. 향후에는 이를 개선하기 위한 연구 및 실험을 통하여, 본 논문에서 제안한 알고리즘을 보다 보편적이고 실질적으로 사용할 수 있도록 개발하도록 할 것이다.

참고문헌

[1] 이상운, 주행시간 기반 실시간 점대점 최단경로 탐색 알고리즘, 한국인터넷방송통신학회, 2012

[2] 오병화, 배준성, 양지훈, 남종호, 차량 간 통신을 이용한 유전자 알고리즘 기반 동적 차량 경로 탐색 알고리즘, 한국컴퓨터종합학술대회논문집, 2009

[3] 송광열, 이준웅, 에이스타 알고리즘을 이용한 무인자율주행자동차의 경로 계획, ICROS학술대회, 2012

[4] 최인규, 자동차 네비게이션 시스템 인터페이스 디자인 분석에 관한 연구, 한국멀티미디어학회 추계학술발표논문집, 2001

[5] 김동현, 전자상거래에서의 이동에이전트 시스템 보안에 관한 연구, 한국전자통신학회 추계종합학술대회지 제3권 제2호, 2009

[6] 김선경, 김진상, 임재걸, 임태수, “컴퓨터 알고리즘”, 도서출판 한산, 2012

[7] 정희석, 이종수, 강화학습을 이용한 주행경로 최적화 알고리즘 개발, 한국지능시스템학회 춘계학술대회, 2003

[8] 정태진, 장병탁, 강화학습을 이용한 웹 정보 검색, 한국정보과학회 가을 학술발표논문집, 2001

[9] 곽환주, 박귀태, 강화학습과 유전자 알고리즘을 이용한 이동로봇의 최적경로 탐색, 정보 및 제어 학술대회, 2010

[10] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", MITPress, 2001

[11] Richard Sutton, Andrea Barto, "Reinforcement Learning : An Introduction", BradfordBook, 1998

[12] Tom M. Mitchell, "Machine Learning", McGraw-Hill, 1997

[13] Peter Stefan, Laszlo Monostori, Ferenc Erdelyi, Reinforcement Learning for Solving Shortest-path and Dynamic Scheduling Problems, IWES'01, 2001

[14] http://en.wikipedia.org/wiki/Heuristic