

A*와 유전자 알고리즘을 이용한 효율적인 경로 탐색

강호균* · 최재혁** · 김광백*

*신라대학교 컴퓨터공학과

**신라대학교 컴퓨터교육과

Efficient Path Search using A* and Genetic Algorithm

Ho Kyun Kang* · Jae Hyuk Choi** · Kwang Beak Kim*

*Dept. of Computer Engineering, Silla University

**Dept. of Computer Education

E-mail : gyun0923@naver.com, jhchoi@silla.ac.kr, gbkim@silla.ac.kr

요 약

본 논문에서는 최적화 문제를 해결하는 기법의 하나인 A* 와 유전자 알고리즘을 이용하여 모든 노드를 탐색하여 최적의 경로를 도출하는 최적화 경로 탐색 방법을 제안한다. 경로를 도출하기 위해 A* 알고리즘을 적용하여 출발지 노드로부터 중간 경로 노드까지의 거리를 측정하여 개체를 생성한다. 출력 노드들을 도출하기 위해 생성된 개체를 적합도 함수에 적용하여 적합도를 계산한다. 계산된 적합도 값에 따라 교배를 할 노드 및 교배 지점을 선택한다. 선택된 노드와 교배 지점을 이용하여 개체들을 교배한다. 교배를 통해 새로운 개체를 생성한다. 새로운 개체가 적합도 조건에 만족하면 출력 노드로 도출하고, 다음 출력 노드를 도출하기 위한 출발지 노드로 선택한다. 이러한 과정을 반복하여 모든 출력 노드를 도출한다.

제안된 방법을 경로 탐색 문제를 대상으로 실험한 결과, A* 알고리즘만을 이용한 경우보다 제안된 방법이 경로 탐색 문제에 있어서 최적화된 거리를 기반으로 경로를 탐색하는 것을 확인하였다.

키워드

최적화, 유전자, 개체, 적합도, A*

I. 서 론

A* 알고리즘은 노드와 노드사이의 최적화된 거리를 계산해주는 알고리즘으로 여러 노드에 대해 최적의 경로를 도출할 수 없다[1]. 따라서 본 논문에서는 단순 A* 알고리즘을 적용하여 순차적으로 경로를 탐색하였을 때와 제안된 경로 탐색 방법을 적용하여 최적의 경로를 탐색하였을 때를 비교한다. 제안된 경로 탐색 방법에는 최적화 방법 중의 하나인 유전자 알고리즘을 적용하여 모든 노드들을 최적으로 탐색하도록 한다. 출발지 노드에서 중간 노드들의 거리를 A* 알고리즘으로 계산한 후, 계산된 거리값을 이용하여 개체를 생성한다. 생성된 개체를 유전자 알고리즘의 선택 연산과 교차 연산을 이용하여 새로운 개체를 생성한다. 생성된 개체를 적합도 조건에 적용하여 최적의 경로를 탐색한다.

II. 제안된 경로 탐색 방법

제안된 최적의 경로 탐색 방법은 노드 간의 최적의 경로를 탐색하기 위해 중간 노드와 출발지 노드 및 도착지 노드 간의 거리 값을 A* 알고리즘을 적용하여 최적의 경로를 탐색한다.

III. A* 알고리즘을 이용한 노드간의 거리값 계산

유전자 알고리즘을 적용하기 위해 각 노드 간에 거리를 A* 알고리즘을 이용하여 계산한다. 이때 각 노드간 거리를 계산할 때에는 식 (1)을 적용하여 거리값을 계산한다.

$$f(n) = g(n) + h(n) \quad (1)$$

식(1)에서 n은 경로의 마지막 노드이고 g(n) 은 출발 꼭짓점으로부터 꼭짓점 n까지의 경로 비용

이며 $h(n)$ 은 n 에서 목표 꼭짓점까지의 추정 경로 가중치를 의미한다.

IV. 유전자 알고리즘을 적용한 최적화 경로 탐색

유전자 알고리즘은 문제가 계산 불가능할 정도로 지나치게 복잡할 경우에는 유전자 알고리즘을 적용하여, 실제 최적해를 구하지는 못하더라도 최적해에 가까운 답을 얻기 위한 방안으로써 접근할 수 있다[2]. 이 경우 해당 문제를 해결하는 데 최적화되어 있는 알고리즘보다 좋은 성능을 보여주지는 못하지만, 대부분 받아들일 수 있는 수준의 해를 보여줄 수 있다[3].

4.1 개체생성

최적의 경로를 탐색하기 위해 각 노드간의 거리를 가중치 값으로 계산하여 유전자 알고리즘의 개체를 생성한다. 이때 개체를 생성할 때에는 다음과 같은 식(1)에 적용하여 가중치 값, 즉 개체를 생성한다. 식(1)을 적용하여 임의로 생성된 개체들을 이용하여 다음 세대 개체들의 적합도 평가 기준을 식(2)와 같이 설정한다.

$$f(x) = O_{Max} * \alpha \quad (2)$$

식(2)의 O_{Max} 는 생성된 개체들의 값 중에서 최대값이고, α 는 적합도 함수의 파라미터로서 초기에는 2.0으로 설정한다. 개체가 적합도의 평가 기준을 만족하지 못한 상태에서 제안된 유전자 알고리즘을 반복하면 적합도의 기준을 감소시킨다. 따라서 평가 기준을 개체들에 맞추도록 하기 위해 α 의 값을 0.1씩 감소시켜 적합도의 평가 기준을 설정한다.

4.2 개체선택

제안된 방법에서는 개체들의 값이 증가할수록 최적의 해를 구하는 환경에서 유리하도록 설정하였다. 따라서 식(3)을 개체들에 적용하여 적합도를 계산한다.

$$r(x) = O_{Max}^2 - x \quad (3)$$

식(3)의 O_{Max} 는 생성된 개체들의 값 중에서 최대값이고, x 는 개체들의 값이다. 식(3)을 적용하여 계산된 값들 중에서 적합도가 가장 높은 상위 2개의 염색체를 보존하고 다음 세대를 생산할 것인지, 아니면 세대를 생성하지 않고 도태될 것인지

에 대한 기준은 식(4)를 적용하여 확률적으로 결정한다.

$$p(x) = r(x) / TN \quad (4)$$

식(4)에서 TN은 식(3)을 계산한 $r(x)$ 값들의 총합이다. $r(x) / TN$ 의 값에 반올림하여 다음 세대를 생산할 수 있는 개체와 도태되어질 개체를 확률적으로 선택한다.

4.3 교차 연산

제안된 방법에서는 모든 해에 대해 교차 연산을 수행하지 않고 개체의 2/3 이후지점에 대해서만 교차 연산을 실행한다.

4.4 유전자 변이

전형적인 돌연변이 확률은 0.05 이하로 설정하는 것이 일반적이다[4]. 따라서 제안된 방법에서는 0.05%확률로 변이 연산을 수행하여 지역 최적해에 위치하는 경우를 방지하였다.

V. 실험 및 결과 분석

본 논문에서 제안한 방법을 Intel(R) Core(TM) i5 CPU @ 2.80GHz 와 8.00GB RAM이 장착된 PC 상에서 Visual Studio 2013 C#으로 구현하여 실험하였다. 그림 1은 단순 A* 방법과 제안된 방법, 랜덤 방법 간의 경로 탐색 결과이다.



그림 1. 순차방법, 제안된 방법과 랜덤방법의 경로 탐색 비교 결과

표 1은 A* 방법을 적용하여 순차방법의 경우와 랜덤방법의 경우 및 제안한 방법에서 도출된 거리 값을 각각 나타내었다. 표 1에서 알 수 있듯이 제안한 방법이 A* 알고리즘을 이용한 방법과 랜덤방법 보다 총 탐색 거리 값이 적으므로 경로 탐색에 있어서 효율적인 것을 확인할 수 있다.

표 1. 순차 방법과 제안된 방법 간의 총 탐색
거리 비교

A* 탐색방법	제안한 방법	랜덤 방법
Index		
0 - 1	0 - 2	0 - 1
- 2 -	- 3 - 4	- 3 -
3 - 4	- 1	4 - 2
총 탐색 거리 값		
865	419	729

VI. 결론 및 향후 개선 방향

본 논문에서는 출발지 노드부터 모든 노드들을 탐색하면서 최적의 경로를 탐색하는 방법을 제안하였다. 최적의 경로를 탐색하기 위해 최적화 방법 중의 하나인 유전자 알고리즘을 적용하였다.

향후 연구 방향은 현재에 적용된 A* 알고리즘 대신에 Iterative deepening A* (IDA*) 알고리즘과 유전자 알고리즘을 적용하여 현재의 성능보다 개선하여 내비게이션에 적용할 수 있도록 할 것이다.

참고문헌

- [1] <http://ko.wikipedia.org/>
- [2] 김광백, 송두현, “유전자 알고리즘을 이용한 경로 탐색” 한국정보통신학회, 한국해양정보통신학회논문지, 제15권 제6호, pp.1251-1255, 2011.
- [3] <http://www.policyalmanac.org/games/aStarTutorial.htm>
- [4] 서민관, 이재성, 김대원, “유전자 알고리즘의 수렴 속도 향상을 통한 효과적인 로봇 길 찾기 알고리즘,” 한국컴퓨터정보학회논문지, 20권, 4호, pp.25-32, 2005.