

유전자 알고리즘을 이용한 효율적인 경로 탐색

조현학[○], 조재현^{**}, 김광백^{*}

[○]신라대학교 컴퓨터정보공학부

^{**}부산가톨릭대학교 컴퓨터공학과

e-mail: darkruby1004@naver.com, jhcho@cup.ac.kr, gbkim@silla.ac.kr

Efficient Path Search Method using Genetic Algorithm

Hyun-Hak Cho[○], Jae-Hyun Cho^{**} and Kwang-Baek Kim^{*}

[○]Division of Computer and Information Engineering, Silla University

^{**}Dept. of Computer Engineering, Catholic University of Pusan

● 요약 ●

본 논문에서는 최적화 문제를 해결하는 기법의 하나인 유전자 알고리즘을 이용하여 모든 노드를 탐색하여 최적의 경로를 도출하는 최적화 경로 탐색 알고리즘을 제안한다. 경로를 도출하기 위해 중간 경로 노드로부터 출발지 노드 및 도착지 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 지점(비트단위)을 선택한다. 선택되어진 교배와 교배 지점을 이용하여 개체들을 교배한다. 교배를 통해 새로운 개체를 생성한다. 새로운 개체가 적합도 조건(이전 개체 중 최대값 * 2)에 만족한다면 출력 노드로 도출하고, 다음 출력 노드를 도출할 때의 출발지 노드로 선택되어진다. 이러한 과정을 반복하여 모든 출력 노드를 도출한다. 제안된 방법을 실험한 결과, 순차 방식과 난수를 이용한 경우보다 제안된 방법이 효율적인 것을 확인하였다.

키워드: 최적화, 유전자, 개체, 적합도

I. 서론

정보 공학에서 최적(화)란, 사용 중인 시스템을 변경 또는 수정하여 작업을 더 효과적으로, 또는 자원을 효율적으로 사용하도록 만드는 작업을 뜻한다. 이를테면, 컴퓨터 소프트웨어를 더 빠르게 실행시키거나 기억 장치나 자원을 덜 차지하게 하여 운영하거나, 전력을 덜 쓰게 함으로써 최적화할 수 있다[1]. 최적의 경로를 찾기 위한 방법으로 최적화 방법 중의 하나인 유전자 알고리즘[2]을 적용하여 모든 노드들을 최적으로 탐색하기 위한 방법을 제안한다. 출발지 노드 및 도착지 노드와 중간 노드들의 유클리디안 거리를 계산한 후 계산된 거리값을 이용하여 개체를 생성한다. 생성된 개체를 유전자 알고리즘의 선택 연산과 교차 연산을 이용하여 새로운 개체를 생성한다. 생성된 개체를 적합도 조건에 대입하여 중간 경로 노드를 도출하여 최적의 경로를 탐색한다.

2.1 개체 생성

최적의 경로를 탐색하기 위해 각 노드 간에 유클리디안 거리를 가중치 값으로 계산하여 유전자 알고리즘의 개체를 생성한다. 이 때 개체를 생성할 때에는 다음과 같은 식(1)에 적용하여 가중치 값, 즉 개체를 생성한다.

$$f(x) = A - SRp + ERp \quad (1)$$

여기서 A는 최대 거리 값이고 SRp는 출발지 노드와 중간 노드의 유클리디안 거리이다. 그리고 ERp는 도착지 노드와 중간 노드의 유클리디안 거리를 의미한다.

현재 생성된 개체들을 이용하여 다음 세대 개체들의 적합도 평가 기준을 식(2)와 같이 설정한다.

$$f(x) = Max * Percent \quad (2)$$

식(2)의 Max는 생성된 개체들의 값 중에서 최대값이고, Percent의 초기값은 2.0으로 설정하였다. 개체가 적합도의 평가 기준을 만족하지 못한 상태에서 제안된 알고리즘을 반복하면 적합도의 기준을 감소시킨다. 따라서 평가 기준을 개체들에 맞추도록 Percent의 값을

II. 제안된 경로 탐색 방법

제안된 최적의 경로 탐색 방법은 노드 간의 최적의 경로를 탐색하기 위해 중간 노드와 출발지 노드 및 도착지 노드 간의 유클리디안 거리 값을 가중치 값으로 이용하여 최적의 경로를 탐색한다.

0.1씩 감소시켜 적합도의 평가기준을 설정한다.

2.2 개체 선택

유전자 알고리즘에서 선택의 문제는 중요한 문제이다. 유전자 알고리즘에서 개체 선택 방법에 따라 최적의 해로 다가가는 속도가 느려지거나, 아니면 지역최소화에 쉽게 빠질 수 있다. 또한, 우수한 개체가 보유한 적합도가 낮은 유전자를 다음 세대 개체들에게 배정할 수도 있다. 이러한 부분을 제어하기 위해 일반적으로 가장 적합한 개체의 순으로 선택될 확률을 높게 부여하는 방법론이 많이 적용되고 있다[3,4].

식(3)과 같이 적합도 함수는 특정 해가 얼마나 적합한지 나타내는 함수로 개체가 유전자에 따라 특정 환경에서 살아남을 수 있는 확률이 존재하는 것과 같다. 적합도 함수는 주어진 해가 다음 세대를 생산할 수 있을지 혹은 세대를 남기지 못하고 도태되어 제거될지를 결정한다[5].

제안된 방법에서는 개체들의 값이 클수록 최적의 해를 구하는 환경에서 유리하도록 설정하였다. 따라서 다음과 같은 식(3)에 개체들을 적용하여 적합도를 계산한다.

$$f(x) = x^2 \quad (3)$$

식(3)의 x는 개체들의 값이다. 식(3)을 적용하여 계산된 값들은 다음 세대를 생산할 수 있을지, 혹은 세대를 남기지 못하고 도태되어 사라질지를 다음과 같은 식(4)를 적용하여 결정한다.

$$f(x) = N / TN \quad (4)$$

식(4)의 N은 식(3)을 계산한 값이고, TN은 식(3)을 계산한 값들의 총합이다. N / TN의 값에 반올림하여 다음 세대를 생산할 수 있는 개체와 도태되어질 개체를 선택하여 제거한다.

2.3 교차 연산

개체는 세대 내에서의 교배를 통해 다음 세대를 생성한다. 일반적으로 두 개의 개체가 교배를 통하여 다음 세대의 개체를 생성하며, 새로운 개체는 각각의 부모 개체로부터 서로 겹치지 않는 위치의 유전자를 받아 새로운 유전자를 구성한다. 이 때 특정 개체간의 교배를 하면 2-2(선택)에서 의미한 바와 같이 우수한 개체들이 가지는 적합도가 낮은 유전자를 다음 세대에 넘겨줄 수 있고, 또한 열등한 개체들이 가지는 적합도가 높은 유전자를 다음 세대에 넘겨 줄 수 없는 오류가 발생한다. 따라서 이러한 부분을 제어하기 위해 난수를 이용하여 교배를 할 개체를 선택한다. 그리고 교배를 한 유전자 위치 역시 난수를 이용하여 교배의 위치를 선택하여 교배를 한다.

III. 실험 및 결과

본 논문에서 제안한 방법을 Intel Core(TM)2 Duo 2.66GHz CPU와 2GB RAM이 장착된 IBM 호환 PC 상에서 Java - NetBeans를 이용해 구현하여 실험하였다.

제안된 방법은 계산된 유클리디안 거리 값을 이용하여 개체를 생성한 후, 선택 연산을 이용하여 적합도를 계산하고 교배 연산에 적용하여 새로운 개체를 생성하였다. 새로 생성된 개체를 적합도 조건에 적용하여 만족할 경우 중간 경로 노드로 도출하였다. 도출 과정에서 적합도 조건에 만족한 개체의 index를 이용하여 중간 경로 노드로서 선택했다. index를 이용한 부분은 본 논문에서 사용한 유전자 알고리즘에서 알 수 있듯이 적합도가 높은 개체일수록 다음 세대를 생성할 수 있는 확률을 높게 설정하였다. 개체가 중간 경로 노드로 도출되었다는 것은 부모 개체가 우수한 개체, 즉 가장 적합한 개체라는 것으로 가정하여 도출된 개체의 index를 이용하여 중간 경로로 선택하였다. 그림 1은 제안된 방법을 이용한 경로 탐색 결과이다.

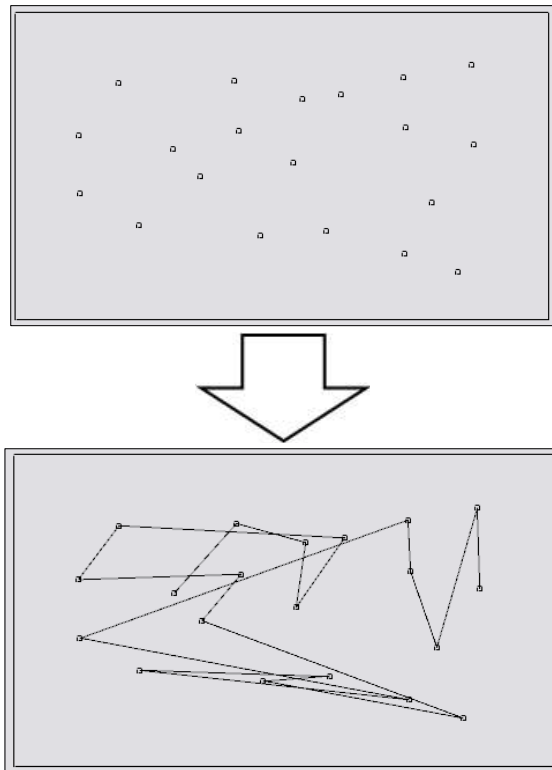


그림 1. 경로 탐색 결과

표 1은 순차 및 난수를 이용한 경우의 거리 값과 제안한 방법의 결과를 나타내었다. 표 1에서 알 수 있듯이 제안한 방법이 순차를 이용한 방법과 난수를 이용한 방법보다 총 탐색 거리 값이 적으므로 경로 탐색에 있어서 효율적인 것을 확인할 수 있다.

표 1. 순차 및 난수 방법과 제안된 방법 간의 총 탐색 거리 비교

순차를 이용한 방법	난수를 이용한 방법	제안한 방법
Index		
1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12 - 13 - 14 - 15 - 16 - 17 - 18 - 19 -20	1 - 19 - 12 - 18 - 8 - 16- 13 - 2 - 4 - 15 - 10 - 7 - 9 - 6 - 17 - 14 - 11 - 5 - 3 - 20	1 - 11 - 16 - 3 - 7 - 12 - 13 - 2 - 19 - 6 - 8 - 18 - 5 - 15 - 14 - 17 - 10 - 9 - 4 - 20
총 탐색 거리 값		
2884,3299	2752,7569	2501,2429

IV. 결론

출발지 노드부터 도착지 노드까지의 모든 노드들을 탐색하면서 최적의 경로를 탐색하는 방법을 제안하였다. 최적의 경로를 탐색하기 위해 최적화 방법 중의 하나인 유전자 알고리즘을 적용하였다. 제안하는 방법은 최적의 경로를 탐색하기 위해 가중치 값으로 출발지 노드 및 도착지 노드와 중간 노드의 유클라디안 거리를 이용하였다. 계산된 유클라디안 거리값을 이용하여 개체를 생성하였고 선택 연산과 교차 연산을 적용하여 우수한 개체를 생성하여 적

합도 조건 검사와 비교한 후, 중간 경로 노드를 도출하였다. 실험에서 알 수 있듯이 제안된 방법이 최적의 거리를 탐색하는데 있어서 효율적인 것을 확인하였다.

향후 연구 방향은 제안된 방법과 불쓰만 머신과 A* 알고리즘과 비교 분석할 것이다.

참고문헌

- [1] <http://ko.wikipedia.org/>
- [2] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag Berlin Heidelberg, 1994.
- [3] 홍석미, 이영아, 정태충, “순회 판매원 문제에서 개미 군락 시스템을 이용한 효율적인 경로 탐색,” 정보과학회논문지 : 소프트웨어 및 응용, 제30권 제9-10호, pp. 862-866, 2003.10.
- [4] 서동일, 문병로, “서열순서화문제를 위한 상위정보를 이용하는 혼합형 유전알고리즘,” 퍼지 및 지능 시스템학회 논문지 제15권 제6호, pp.661-667, 2005.12.
- [5] 조재훈, 이대중, 송창규, 김용삼, 전명근, “유전자 알고리즘과 정보이론을 이용한 속성 선택,” 한국지능시스템학회 논문지, 18권, 1호, pp.94-99, 2008.2.