

# 막힌 부분을 갖는 네트워크 내 효과적인 경로 탐색을 위한 유전 알고리즘 적용

김준우\*, 이민정\*\*

요약

자연계의 진화 과정을 모방하는 유전 알고리즘은 다양한 조합 최적화와 같은 NP-hard 문제의 해를 탐색하는데 매우 유용한 도구이다. 본 논문은 네트워크 내에 존재하는 두 노드 사이의 최단 경로를 구하는 문제 풀이를 위하여 유전 알고리즘을 적용하고자 하며, 특히 실행가능한 경로가 최소한 네트워크를 다룬다. 제안하는 유전 알고리즘은 적절히 선별된 노드의 다음 방문 노드를 명시하여 해를 표현하며, 별도의 복구 작업을 포함하지 않는다. 이러한 표현 방법을 사용할 경우, 해 집단 내에 실행불가능한 경로나 순환이 있는 경로가 포함될 수 있으나, 이러한 경로들은 적절히 설계된 적합도 함수에 의해 처리된다. 결과적으로, 제안하는 유전 알고리즘은 순환이나 막다른 노드가 많은 네트워크에서 효과적이고 실행가능한 경로를 탐색하는데 특히 유용할 것으로 생각된다. 잘 알려진 강 건너기 문제 중 하나인 선교사와 식인종 문제를 통해 실험한 결과, 제안하는 유전 알고리즘을 통해 전체 구조를 분석하기 어려운 네트워크에 속한 두 노드 간의 경로를 효과적으로 구할 수 있음을 알 수 있었다.

## Applying Genetic Algorithm for Exploring the Effective Path in Network with Dead Ends

Jun Woo Kim\*, Min Jung Lee\*\*

ABSTRACT

The genetic algorithms that mimic the mechanism of the natural evolution is very useful tool for exploring the solutions of the NP-hard problems such as a variety of combinatorial optimization problems. This paper aims to apply the genetic algorithm to solve the shortest path problem, and especially, the network with few feasible paths is considered. The proposed algorithm identifies the relevant nodes at first, and represents a solution by specifying their adjacent nodes to visit next. Moreover, any additional repairing procedure is considered. This simple encoding can make the population to include infeasible paths or paths with cycles, however, such paths can be deal with by the properly designed fitness function. Consequently, it is expected that the proposed genetic algorithm will be useful to explore the effective feasible paths in the networks with cycles or dead-end nodes, especially. The result of the experiment with the missionaries and cannibals problem, a well-known river-crossing problem, reveals that a path between two nodes in a network of which entire structure is hard to be analyzed can be obtained effectively by the proposed genetic algorithm.

Key Words : Shortest Path Problem, Maze Passing Problem, Combinatorial Optimization, Genetic Algorithm, Missionaries and Cannibals Problem

---

\* 동아대학교 산업경영공학과(✉kjunwoo@dau.ac.kr)

\*\* 세종사이버대학교 경영학과

· 제1저자(First Author) : 김준우 · 교신저자(Correspondent Author) : 이민정

· 접수일(2013년 1월 8일), 수정일(1차 : 2013년 2월 15일), 게재확정일(2013년 2월 18일)

## 1. 서 론

경로 계획(path planning)은 여러 개의 노드와 그 노드들 간의 연결로 구성된 네트워크에서 특정 목적을 충족할 수 있는 경로를 탐색하는 것으로, 최근 차량이나 항공기와 같은 이동 수단의 이동 경로나 이동통신 망 내의 데이터 전달 경로를 결정하는 효과적인 수단이 된다[1][2]. 아울러, 네트워크 안 특정 두 개 노드를 연결하는 가장 짧은 경로를 찾는 문제를 최단 경로 문제(shortest path problem)라고 한다. 이러한 최단 경로 문제는 시작 노드에서 출발하여 도중에 만나는 노드들에서 일련의 이산적인 의사결정을 수행하여 해를 생성할 수 있다는 점에서 조합 최적화(combinatorial optimization) 문제의 하나로 볼 수 있다[3].

최단 경로 문제와 유사한 문제로는 미로 통과(maze passing) 문제가 있다[4]. 미로 통과 문제는 주어진 네트워크 내 두 개 노드 간의 효과적인 경로를 탐색한다는 점에서 최단 경로 문제와 유사하나, 일반적으로 실행가능한 경로를 형성하는 것이 최단 경로 문제보다 어렵고, 종종 실행가능한 경로를 찾는 것 자체가 목적이 된다.

유전 알고리즘은 자연계에서 관찰되는 생체의 진화 과정을 모방하여, 복수의 해들을 유지하면서 이들을 진화시켜가며 좋은 해를 탐색해나가는 방법으로, 이러한 조합 최적화 문제를 푸는데 적합하다[5]. 더불어, 최단 경로 문제나 미로 문제와 같은 경로 계획에 있어서도 유전 알고리즘이 다양하게 적용되어 좋은 성과를 얻어 왔다 [1][4][6][7].

하지만, 일반적으로 문제의 특성에 따라 유전 알고리즘의 세부적인 절차나 특성은 조금씩 달라지고, 이에 따라 특정 형태의 문제에 맞추어 개발된 유전 알고리즘을 다른 형태의 문제에 적용하기 위해서는 어느 정도의 수정이 필요한 경

우가 일반적이다. 이러한 점은 최단 경로 문제와 미로 통과 문제에 적용되었던 유전 알고리즘에서도 관찰된다.

본 논문에서는 실행가능한 경로들 중 최단 거리를 갖는 경로를 탐색하는데 초점을 맞추는 최단 경로 문제와 실행가능한 경로 탐색 자체에 보다 초점을 맞추는 미로 탐색 문제의 특성을 모두 고려할 수 있는 간단한 유전 알고리즘을 제안하고자 한다. 이러한 유전 알고리즘을 통해 미로처럼 막다른 노드나 순환이 자주 발생하는 네트워크에서도 효과적인 경로를 찾아낼 수 있으며, 제안하는 유전 알고리즘을 잘 알려진 강 건너기(river crossing) 문제인 선교사와 식인종(missionaries and cannibals) 문제[8]에 적용한 결과, 효과적으로 좋은 경로를 탐색하는 것을 볼 수 있었다.

본 논문의 2장에서는 제안하는 유전 알고리즘의 연구 배경을 소개하고, 3장에서 이 유전 알고리즘의 세부적인 특성을 설명한다. 4장에서는 선교사와 식인종 문제에 이를 적용한 결과를 소개하며, 끝으로 5장에서 결론 및 추후 과제를 제시한다.

## II. 연구 배경

네트워크는 여러 개의 노드들과 노드들 간의 연결로 이루어진 구조를 의미하며, 통신장비들로 구성된 통신망, 경유지들로 구성된 교통망에서 현실적으로 관찰할 수 있다. 경로 계획은 네트워크 내에 존재하는 효과적인 경로를 탐색하는데 목적을 둔다[2].

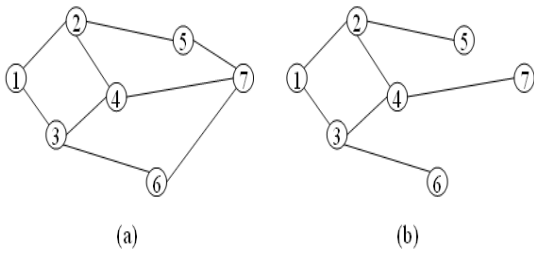


그림 1. 7개 노드를 갖는 네트워크  
Fig. 1. A Network with 7 Nodes

경로 계획 중, 주어진 두 개 노드 간의 경로를 탐색하는 문제로 최단 경로 문제와 미로 통과 문제를 들 수 있다. 두 문제가 명확히 구분되는 것은 아니지만, 일반적으로 각 문제에서 다루어지는 네트워크의 특성에서 차이점을 찾아볼 수 있다. 예를 들어 <그림 1>의 네트워크의 노드 1에서 노드 7까지의 경로를 찾는다고 가정하자. 최단 경로 문제는 보통 (a)에서처럼 각 노드들이 완전 연결(fully-connected) 상태는 아니더라도 연결 경로가 비교적 많은 네트워크를 다룬다.

반면, 미로 문제에서는 (b)에서처럼 잘못된 경로를 따를 경우, 노드 5나 6처럼 막다른 곳에 이르게 되기 쉬운 네트워크를 다룬다. 이로 인하여, 최단 경로 문제는 일반적으로 실행가능 경로를 만드는 것이 용이하며, 이러한 실행가능 경로 중 최단 거리인 것을 탐색하게 된다. 반면, 미로 통과 문제에서는 실행가능 경로를 만드는 것이 비교적 어렵고, 실행가능 경로가 유일하거나, 실행경로를 찾는 것 자체가 목적이 된다. 이렇게 네트워크 상에 존재하는 경로를 탐색하는 문제들은 조합 최적화 문제에 해당하고, 유전 알고리즘을 통해 효과적인 풀이가 가능하다.

최단 경로 문제를 유전 알고리즘으로 풀이할 때는 일반적으로 적절한 유전 연산자나 복구 작업을 통해 실행가능한 경로들로 구성된 해 집단을 유지하면서 이들 중 최단 경로인 것을 찾게 된다. 세부적으로는 이러한 목적을 위하여 다양한 해 표현(encoding) 방법이 사용되어 왔으며, 우선 순위 기반 표현[1], 노드들의 나

열을 이용한 가변 길이 표현[2], 각 노드에서 선택하는 다음 노드들을 이용한 표현[7][9] 등이 사용되었다. 그러나 이러한 방법들은 네트워크 내에 순환이나 막다른 곳이 많이 존재할 경우, 복구 작업에 걸리는 시간이 많아진다는 단점이 존재한다.

반면, 미로 통과 문제는 일반적으로 <그림 1>과 같은 네트워크가 주어지기보다 <그림 2>와 같이 평면의 격자로 미로가 주어지는 경우가 많다.

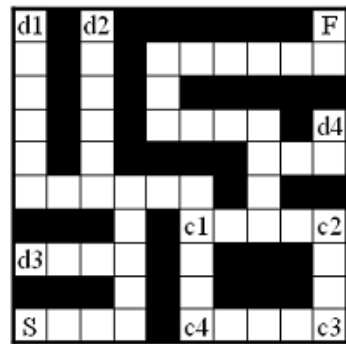


그림 2. 미로 문제  
Fig. 2. The Maze Passing Problem

예를 들어, <그림 2>의 미로는 10×10 격자 위에 장애물에 해당하는 칸을 색칠하여 나타내었고, 격자 S에서 격자 F까지의 경로를 탐색하고자 한다. 미로 문제의 특성 상, 격자 d1~d4와 같은 막다른 곳이 여러 개 존재하며, 격자 c1~c4와 같이 이동 경로에 따라 순환이 발생할 수도 있다. 미로 문제를 위한 유전 알고리즘에서는 보통 각 격자에서의 이동 방향을 명시하여 해를 표현하는 방법을 사용한다[10][11][12].

이는 최단 경로 문제에서 사용되는 방법 중, 각 노드에서 다음에 방문할 노드들을 명시하는 방법과 유사하며, 예를 들어, <그림 2>에서는 이동가능한 흰색 격자가 총 57개 존재하고, 각 격자에서 상, 하, 좌, 우의 4가지 이동 방향을 허용할 경우, 총 4<sup>57</sup>가지 해가 존재한다. 또한, 문제의 특성 상, 해 집단 내에 막다른 곳에

다다르거나 순환에 이르는 실행불가능한 해들이 유지된다는 특성이 있다. 이러한 방법은 유전자의 길이가 길어진다는 점과 해의 실행불가능한 해를 탐색하는 횟수가 많아진다는 단점이 있다.

### III. 경로 탐색을 위한 유전 알고리즘

#### 3.1 해의 표현과 초기해 생성

본 논문에서는 최단 경로 문제와 미로 문제의 장점들을 결합시킨 유전 알고리즘 기반 경로 탐색 전략을 제안하고자 한다. 이를 위하여 먼저 주어진 네트워크에서 유전자로 포함될 노드들을 선별하는 절차를 거친다.

선별 과정에서는 먼저 막다른 노드에 해당하는 것들이 배제되고, 남은 노드들에 대하여 인접 노드들을 조사하는데, 이 때 인접 노드 중 막다른 노드들은 배제한다. 남은 노드들 중에서는 다시 인접 노드가 단 1개인 것들을 제외시키고, 최종적으로 남은 각 노드들에 대하여 유전자가 한 개씩 할당된다. 아울러, 설명의 편의를 위해 각 노드들 간의 거리는 1로 가정한다.

locus	1	2	3	4	...	N
Gene	G <sub>1</sub>	G <sub>2</sub>	G <sub>3</sub>	G <sub>4</sub>	...	G <sub>N</sub>

그림 3. 유전형 표현

Fig. 3. Genetic Representation

예를 들어, 막다른 노드가 제거된  $N$ 개의 노드를 집합  $V = \{Nod_i, i = 1, 2, \dots, N\}$  라 할 때, 각  $Nod_i$ 에 대한 인접 노드  $n_i$ 개의 집합을  $aV_i = \{aNod_{ij}, j = 1, 2, \dots, n_i\}$  라 쓸 수 있다.

개별 해는 기존의 다음 방문 노드 기반 표현과

유사하게 나타나며, <그림 3>과 같이 총  $N$ 개의 유전자를 갖게 된다. 위치  $i$ 의 유전자 값  $G_i$ 는  $Nod_i$ 에 도착 후 다음에 이동할 노드를 의미하며,  $G_i \in aV_i$ 를 만족해야 한다. 초기 해 집단 생성 시에는 각 위치  $i(=1, 2, \dots, N)$ 에서  $aV_i$ 의 원소들 중, 임의의 한 개를 선택하는 것을 반복하면 된다.

#### 3.2 해의 평가와 적합도

앞의 과정에서 막다른 노드들을 미리 노드 집합에서 배제하였으나, 여전히 표현된 경로들 중에서는 실행불가능한 것들이 많이 존재할 수 있다. 그 이유로는 첫 번째, 기존의 막다른 노드가 제거되었더라도, 제거된 노드의 인접 노드  $Nod_k$ 의 남은 인접 노드가  $V$ 에  $Nod_0$  단 하나밖에 존재하지 않고, 동시에 시작점에서  $Nod_0$ 를 거쳐야만  $Nod_k$ 에 도달할 수 있는 경우,  $Nod_k$ 는 새로운 막다른 노드가 된다. 더불어,  $Nod_i (i = 1, 2, \dots, N)$ 의 다음 방문 노드  $G_i$ 들에 의한 순회 시, 순환이 야기되어 목적지에 도달할 수 없을 수도 있다. 본 논문에서 제안하는 유전 알고리즘에서는 이러한 점들을 별도의 복구 작업을 통해 해결하기보다 적절히 설계된 적합도 함수를 통해 처리하며, 해 집단 내의 해  $c_p (p = 1, 2, \dots, P)$ 의 적합도  $F(p)$ 는 (1)과 같이 산출된다. (단,  $P$ 는 해집단의 크기를 의미한다.)

$$F(p) = \begin{cases} \frac{K}{s(p)}, & \text{실행가능한 경우} \\ \frac{K}{s(p) + a(N-s(p)) + bN}, & \text{기타} \end{cases} \quad (1)$$

표 1.  $s(p)$  산출 과정

Table 1. Computation Procedure for  $s(p)$

```

set 각 노드 방문 횟수 = 0
set state = 시작 노드
set  $s(p) = 1$ 
set pre_state = 0
set isPath = false

while(true)
{
  if (state = 종료노드)
    isPath = true
    중단
  else
  {
    pre_state = state
    state = Gene(p, locus(state))

    if (state 방문 횟수=1)
      중단
    else
    {
      s(p)++
      state 방문 횟수++
    }
  }
}

```

(1)에서  $K$ 는 상수를 의미하며,  $s(p)$ 는  $c_p$ 에 의해 경로를 탐색했을 때, 방문하게 되는 노드 개수이다. (단, 각 노드 간 거리가 서로 다른 경우에는  $s(p)$ 는 이동하게 되는 총 거리를 나타낸다.) 즉,  $c_p$ 가 시작 노드에서 종료 노드까지 탐색이 가능한 경로를 의미한다면, 그 적합도는 이동 횟수(거리)에 반비례하며, 이동 횟수  $s(p)$ 는 <표 1>과 같이 산출한다. 단,  $locus(Nod_1)$ 는 염색체에서  $Nod_1$ 에 해당하는 위치를 의미하고,  $Gene(a, b)$ 는 해 집단 내 개체  $c_a$ 의  $b$ 번째 유전자 값을 의미한다.

$s(p)$ 를 산출할 때 시작 노드에서 출발하여 매 방문 노드마다 이동 횟수를 집계하고, 도중에 기존에 방문했던 노드를 다시 방문할 경우, 막다른 노드나 순환이 발생한 것으로 보아 탐색을 중단한다.

반면, 개체  $c_p$ 에 의한 탐색 결과 종료 노드에 도달하지 못하는 경우에는 (1)의 아래쪽 식에 의하여 적합도가 계산된다. 이 경우에는 종료 노드까지 도달한 것과 마찬가지로 분모에  $s(p)$ 가 포함되고 아울러 두 가지의 별점 항목이 추가된다. 첫 번째 별점 항목은 실행불가능한 경로를 의미하는 해 중에서도 많은 노드를 탐색한 것을 우대하는 것으로 미방문 노드의 개수  $N-s(p)$ 에 계수  $a$ 를 곱하여 산출한다. 두 번째 별점 항목은 실행불가능한 점 자체에 대한 항목으로 노드 개수  $N$ 에 계수  $b$ 가 곱해진다. 계수  $a$ 와  $b$ 를 적절히 조절하여 유전 알고리즘 실행 도중의 선택압력(selection pressure)을 조정할 수 있으며, 계수들의 값을 부여할 때는 실행불가능한 경로들이 실행가능한 경로보다 낮은 적합도를 갖도록 하는 것이 바람직하다.

결과적으로 이러한 방법을 통해 막다른 노드나 순환을 많이 갖는 네트워크의 노드 개수를 어느 정도 감소시킨 후, 별도 복구 작업 없이 네트워크에서 좋은 경로를 탐색해나가는 것이 가능하다.

### 3.3 유전 연산자

본 논문에서 제안하는 유전 알고리즘에서 선택 연산 방법으로는 일반적인 확률바퀴 선택을 사용하며, 교차 방법은 일양교차를 적용하였다. 돌연변이 연산은 개별 유전자 단위로 시행되며, 위치  $i$ 의 유전자에서 돌연변이 연산이 발생하는 경우, 집합  $aV_i$ 의 원소 중, 현재 유전자 값과 다른 것을 임의로 한 개 선택하도록 하였다.

## IV. 실험 결과

### 4.1 선교사와 식인종 문제

선교사와 식인종 문제는 일종의 퍼즐로, 처음에 강 한쪽에 선교사와 식인종 각 3명과 두 명까지 탈 수 있는 배가 존재한다. 게임의 목표는 모든 사람을 강의 반대쪽으로 이동시키는 것인데, 어느 쪽이든 식인종의 수가 선교사보다 많아지면 게임이 실패로 종료된다.

표 2. 선교사와 식인종 문제 내 노드  
Table 2. Nodes in the Missionaries and Cannibals Problem

번호	선교사	식인종	배	인접노드
1	0	0	O	-
2	0	1	O	11
3	0	2	O	11
4	0	3	O	12, 13
5	1	1	O	11
6	2	2	O	13, 15
7	3	0	O	-
8	3	1	O	15, 17
9	3	2	O	16, 17, 18
10	3	3	O	16, 18, 19
11	0	0	X	-
12	0	1	X	3, 4, 5
13	0	2	X	4, 6
14	0	3	X	-
15	1	1	X	6, 8
16	2	2	X	9
17	3	0	X	8, 9
18	3	1	X	9
19	3	2	X	-
20	3	3	X	-

이 문제의 탐색 공간은 게임의 각 상태를 노드로 하고, 한 번의 조작으로 도달가능한 상태끼리 연결된 네트워크로 귀결되며, 게임 초기 상태와 완료 상태는 각각 시작 노드와 종료 노드가 된다. 아울러, 게임 실패 상태는 막다른 노드가 되며, 시작 노드에서 종료 노드까지의 경로를 찾는 경우 문제가 해결된다.

표 3. 선교사와 식인종 문제를 위한 유전형 표현

Table 3. Genetic Representation for the Missionaries and Cannibals Problem

유전자	의미	가능한 값
1	노드 4의 다음 노드	12, 13
2	노드 6의 다음 노드	13, 15
3	노드 8의 다음 노드	15, 17
4	노드 9의 다음 노드	16, 17, 18
5	노드 10의 다음 노드	16, 18, 19
6	노드 12의 다음 노드	3, 4, 5
7	노드 13의 다음 노드	4, 6
8	노드 15의 다음 노드	6, 8
9	노드 17의 다음 노드	8, 9

Id	1	2	3	4	5	6	7	8	9
1	12	13	15	17	16	3	4	6	8
2	12	13	15	17	16	5	4	6	8
3	12	13	15	17	18	3	4	6	8
4	12	13	15	17	18	5	4	6	8

그림 3. 최적의 경로들

Fig. 3. The Optimal Paths

제안하는 유전 알고리즘을 이 문제에 적용하기 위해서는 먼저, 상태 네트워크에 존재하는 노드 중 막다른 것이 아닌 것들을 찾아야 한다. 처음에 선교사와 식인종들이 있는 쪽을 A, 이동해야 할 반대쪽을 B라고 할 때, A영역에 존재하는 사람 및 배를 기준으로 게임을 종료시키지 않는 상태들을 찾아보면 <표 2>와 같이 총 20개가 있고, 이들 중 10, 11번 노드가 각각 출발 노드, 종료 노드가 된다. 아울러, 각 노드의 인접 노드 집합  $aV_i$  역시 <표 2>의 5번째 열에서 볼 수 있다.

이들 중, 인접 노드가 1개이거나 없는 것들을 제외하면 최종 선별되는 노드는 4, 6, 8, 9, 10, 12, 13, 15, 17 번의 9개로, 염색체 1개는 <표 3>과 같이 총 9개의 유전자를 포함한다. 이들 중 2개 선택지를 갖는 유전자가 6개, 3개 선택지를 갖는 유전자가 3개 존재하기 때문에 이들을 조합하여 만들 수 있는 개체는 총  $2^6 \times 3^3 = 1728$ 가지이다. 참고로, 이 문제의 최적의

경로는 <그림 3>의 4개가 존재한다.

#### 4.2 해 탐색 결과

이 문제의 네트워크 내 경로 탐색을 위하여 유전 파라미터들을 해 집단의 크기  $p=10$ , 교차율  $=0.6$ , 돌연변이율  $=0.05$ 로 설정하고, 적합도 함수 내의 계수들을  $K=100$ ,  $a=3$ ,  $b=1$ 로 사용하기로 하였다. 아울러, 우수 개체 유지를 위해 매 진화 시 가장 좋은 2개의 개체를 보존하도록 하여 문제를 10회 풀이한 결과 평균적으로 20.7회의 반복을 거쳐 최적의 경로 중 하나를 얻을 수 있었다. 반면, 비교를 위하여 우수 개체 유지 전략 없이 10회 풀이했을 때는 최적의 경로를 얻는데 평균적으로 49.1회의 진화가 소요되어, 우수 개체 유지 전략이 경로 탐색에 큰 영향을 미치는 것을 알 수 있었다.

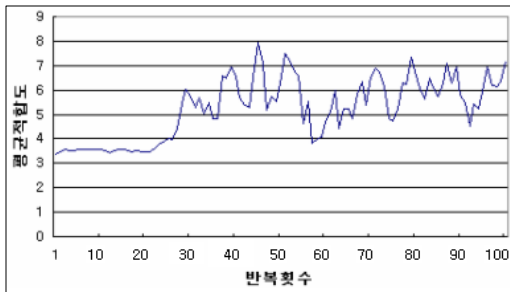


그림 4. 평균 적합도 추이

Fig. 4. The Trend of Average Fitness

한편, 우수 개체 유지 전략을 사용하면서 100회의 진화를 거치는 동안, 해 집단 평균 적합도를 관찰한 결과는 <그림 4>와 같다. 이를 보면, 초기에 평균 적합도가 안정적으로 상승하는 것을 볼 수 있고, 이는 해 집단 내에 실행불가능한 해들이 존재하면서 점진적으로 개선된 해를 찾기 때문으로 생각된다. 반면, 반복횟수 30회를 전후해서 평균 적합도의 변동이 심해지는데, 이는 1개 이상의 실

행가능 경로를 찾은 이후, 유전 연산을 통하여 실행가능 해로부터 실행불가능한 후손이 등장가능하기 때문이며, 향후 보완이 필요할 것으로 보인다.

#### V. 결론 및 추후 연구 과제

본 논문에서 제안된 알고리즘은 종래 최단 경로 문제 풀이와 달리 복잡한 복귀 작업을 요구하지 않고, 종래 미로 통과 문제와 달리 사전 노드 선별을 통해 어느 정도 실행불가능한 해들을 배제한다는 장점이 있다.

아울러, 초기에는 해 집단 내에 비효율적인 경로들이 많이 존재하나 적절히 설계된 적합도 함수를 통해 이들을 처리하고, 진화를 거듭해나가면서 좋은 경로를 찾아내게 된다. 특히, 우수 개체 유지 전략을 사용할 경우 알고리즘이 좋은 성능을 보였다.

하지만 진화 과정에서 실행가능한 해를 찾은 이후 유전 연산을 통해 이들이 다시 파괴되는 경우가 종종 발생하는 것으로 보여, 실행가능 해를 찾은 이후에는 실행가능성을 유지하는 유전 연산자를 적용하는 등의 추가적인 개선이 필요할 것으로 보인다. 아울러, 노드와 연결이 명시적으로 주어지지 않는 미로 통과 문제에 대한 적용 방안도 추후 연구 과제로 남아 있다.

#### 참고문헌

- [1] Gen, M., Cheng, R., and Wang, D., "Genetic Algorithms for Solving Shortest Path Problems," Proceedings of Congress on Evolutionary Computation 2004, pp.835-838.
- [2] Ahn, C.W., and Ramakrishna, R.S., "A Genetic Algorithm for Shortest Path Routing Problem and the Sizing of Populations," IEEE Transactions on Evolutionary Computation, Vol.6, No.6, pp.566-579.
- [3] Hu, Y., Yang, S.X., Xu, L.-Z., and Meng, Q.-H., "A Knowledge Based Genetic Algorithm for Path

Planning in Unstructured Mobile Robot Environments," Proceedings of the 2004 IEEE International Conference on Robotics and Biomimetics, pp.767-772.

- [4] Su, S., and Tsuchiya, K., "Learning of a Maze Using a Genetic Algorithm," Proceedings of the International Conference on Industrial Electronics, Control, and Instrumentation, pp.376-379, 1993.
- [5] Holland, J.H., "Genetic Algorithm," Scientific American, Vol.266, pp.44-50, 1992.
- [6] Elshamli, A., Abhullah, H.A., and Areibi, S., "Genetic Algorithm for Dynamic Path Planning," Proceedings of the Canadian Conference on Electrical and Computer Engineering, pp.677-680, 2004.
- [7] Li, S., Ding, M., Cai, C., and Jiang, L., "Efficient Path Planning Method based on Genetic Algorithm Combining Path Network," Proceedings of the 4th International Conference on Genetic and Evolutionary Computing, pp.194-197, 2010.
- [8] [http://en.wikipedia.org/wiki/Missionaries\\_and\\_cannibals\\_problem](http://en.wikipedia.org/wiki/Missionaries_and_cannibals_problem)
- [9] Inagaki, J., Maseyama, M., and Kitajima, H., "A Genetic Algorithm for Determining Multiple Routes and Its Applications," Proceedings of the IEEE International Symposium on Circuits and Systems, pp.137-140, 1999.
- [10] Carrick, C., and MacLoed, K., "An Evaluation of Genetic Algorithm Solutions in Optimization and Machine Learning," Proceedings of the 21st Annual Conference of Canadian Association for Information Sciences, pp.224-231.
- [11] Baba, N., and Handa, H., "Genetic Algorithm Applied to Maze Passing Problem of Mobile Robot - A Comparison with the Learning Performance of the Hierarchical Structure Stochastic Automata," Proceedings of the 1994 IEEE International Conference on Neural Networks, pp.2690-2695.
- [12] Gordon, V.S., and Matley, Z., "Evolving Sparse Direction Maps for Maze Pathfinding," Proceedings of Congress on Evolutionary Computation 2004, pp.835-838.

## 감사의 글

이 논문은 동아대학교 교내연구비 지원에 의하여 연구되었음.

## 저자소개

### 김준우(Jun-Woo Kim)



2001년 한국과학기술원 산업공학과 졸업(공학사)  
2003년 한국과학기술원 산업공학과 졸업(공학석사)  
2009년 한국과학기술원 산업 및 시스템공학과 졸업(공학박사)

2009년~2010년 한국기술교육대학교 산업경영학부 대우 교수

2011년~현재 동아대학교 산업경영공학과 조교수

※ 관심분야 : 데이터마이닝, 지능형 시스템, 조합 최적화, 메타 휴리스틱, e-러닝, serious game

### 이민정(Min Jung Lee)



1999년 한국과학기술원 재료공학과 졸업(공학사)  
2001년 한국과학기술원 재료공학과 졸업(공학석사)  
2009년 한국과학기술원 산업 및 시스템공학과 졸업(공학박사)

2008년~2009년 한국산업기술진흥원

2010년 엔씨소프트

2011년 연세대학교 연구교수

2012년~현재 세종사이버대학교 경영학과 조교수

※ 관심분야 : MIS, CRM, 품질경영, 기술경영, 데이터마이닝