

Development of Path-planning using Genetic Algorithm

崔 漢 秀* · 鄭 憲**
(Han-Soo Choi · Heon Jeong)

Abstract - In this paper, we propose a new method of path planning for autonomous mobile robot in mapped circumstance. To search the optimal path, we adopt the genetic algorithm which is based on the natural mechanics of selection, crossover and mutation. We propose a method for generating the path population, selection and evaluation in genetic algorithm. Simulations show the efficiency for the global path planning, if we adopt the proposed GA method

Key Words : Global Path Planning, Autonomous Mobile Robot, Genetic Algorithm

1. 서 론

자율 이동 로봇(Autonomous Mobile Robot : AMR) 분야는 공장자동화의 영역에서의 물류 운반, 위험지역내의 감시역할 및 각종 서비스 분야에 적용대상이 잠재되어 있으며, 강인한 시스템과 센서개발, 그리고 신호 및 자료처리 능력의 향상으로 80년대 중반부터 활발한 연구가 진행되어 왔다.[1] 현재 FMS와 같은 자동화 공장에서 공구, 부품과 같은 물류를 운반하는데 사용되는 무인 운반체는 고정된 경로만을 반복 이동하는 고정제도 방식이 주류를 이루고 있다. 그러나 경로상에 장애물이 있는 경우, 그것이 제거될 때까지 기다려야 하며 공장 내부구조의 변경이 필요할 경우 제도를 다시 설치해야 하고, 그에 따른 경비가 많이 소요되는 단점이 있다. 따라서 이와 같은 주변환경 변화에 능동적으로 대처할 수 있는 자율 이동 방식 로봇에 대한 연구의 중요성이 증대되고 있다.

자율 이동 로봇의 연구는 로봇의 위치 추정, 최단경로 탐색, 장애물 충돌 회피, 로봇의 궤적 제어, 센서 정보 융합방법등의 분야가 있다. 로봇이 이동하기 위해서는 로봇의 현재 위치에서 목표점까지의 경로가 설정되어야 하며, 경로 계획은 로봇의 주변환경 정보에 따라 크게 두가지 형태로 구분된다. 활동 영역 안에 있는 장애물의 위치와 모양 등의 정보를 완벽하게 인지하고 경로를 계획하는 전역경로계획(Global Path Planning)과 장애물에 대한 사전지식 없이 경로를 계획하는 지역경로계획(Local Path Planning)이 있다.

전역경로계획은 작업에 필요한 정보를 정확하게 수집하여 작업수행전에 경로를 탐색한다. 이 경우 최적의 경로를 얻을 수 있는 장점이 있는 반면 전역경로계획의 개루프 특성때문

에 변화하는 환경 또는 센서의 궤환이 필요한 환경에서는 사용이 어려운 단점이 있다. 지역경로계획은 센서기반 경로계획 또는 불완전한 정보기반 경로계획이라고 하며, 이동로봇이 주행 환경에 관한 사전지식이 필요 없다는 장점이 있다. 안정되면서 최적의 이동을 하기 위해서는 먼저 주어진 지도에서 최단 전역 경로를 얻고, 얻어진 경로로 이동하면서 예기치 못한 장애물을 만났을 경우 충돌회피를 통해 경로를 계속 유지하는 방법이 필요하다.

본 연구는 전역경로계획 문제를 다룬다. 전역 최단 경로를 탐색하는 방법에는 깊이 우선 탐색 법, DP(Dynamic programming)법, 골격화, 가시도표(Visibility Graph)법[2,3], Voronoi diagram법[4], DT(Distant Transfer)법[5] 및 랜덤탐색법, 셀분해법[6], 인공전위계법[7]등이 있다. 이런 연구들은 각각 나름대로의 효율적인 최단 경로를 제시하여 주지만 많은 시간의 소모 및 국소 최저경로에 수렴해버리는 미비함이 아직 존재한다. 특히 장애물이 거의 없는 경우와 아주 많은 경우에 더욱 그러하다.

이러한 문제점을 해결키 위한 방법으로서 생성된 여러개의 경로 중 최적의 경로를 빠른시간에 안정적으로 찾을 수 있도록 선택과 교차 및 돌연 변이 과정을 통한 적자 생존의 원리가 적용되는 유전자 알고리즘(Genetic Algorithm)을 적용하는 연구가 최근 진행되고 있다.[9,10] 유전자 알고리즘은 지금까지의 방법으로써는 해결이 곤란한 문제에 대해서 실용상의 최적해를 랜덤하고 빠르게 찾는 방법으로서 사용되어 왔으며, 종래의 검색 알고리즘과 달리 검색 공간 중 하나의 점뿐만 아니라 많은 점을 동시에 고려하기 때문에 국부최소점(local minima)점에 수렴되는 현상을 극복할 수 있다. 하지만 기존의 유전자알고리즘을 이용한 전역경로 설정연구는 가능 경로를 미리 연결하고 부적절한 경로를 도태시키는 방법의 접근이 이루어졌으며, 실험결과에서도 오목 다면체나 원형 또는 복잡한 통로와 같은 지도영역에서는 적용이 어려웠다. 따라서 다양하면서도 복잡한 지도에서도 안정적으로 최단경로를 얻을 수 있는 새로운 경로생성 및 진화 알고리즘이 요구되고

* 正 會 員 : 朝鮮大 工大 制御計測工學科 教授 · 工博

** 正 會 員 : 朝鮮大 工大 制御計測工學科 工博

接受日字 : 1998年 10月 9日

最終完了 : 1999年 6月 15日

있다.

본 논문에서는 오목 다면체나 원형 또는 복잡한 통로와 같이 다양하고 복잡한 지도에서도 안정적으로 최단경로를 얻을 수 있는 전역경로설계방법을 제안한다. 제시하는 방법은 유전자알고리즘을 적용하기에 적합하도록 지도를 재맵핑하는 전처리과정으로서 셀분해법과 가시도표법을 수행하며, 유전자 알고리즘의 경로진화방법으로서 기존의 유전자알고리즘에서 사용한 부적합 경로 도태방법과는 달리 새로운 경로를 생성하고 평가후 선택하여 진화시키는 방법을 수행한다. 여러 가지 모의실험을 통하여 제안한 유전자알고리즘은 기존의 유전자 알고리즘에서 비해 경로 탐색 횟수를 줄일 수 있으며 오목 및 좁은 통로의 경우에도 이동로봇이 안정적으로 최단 경로를 설정하는 함을 확인한다.

2. 유전자알고리즘

유전자 알고리즘은 1970년대 초기에 Jone Holland에 의해 제안된 자연 도해의 원리를 바탕으로한 최적화 수법이다.[11] 또한, 유전자 알고리즘은 지금까지의 방법으로서는 해결이 곤란한 문제에 대해서 실용상의 최적해를 랜덤하고 빠르게 찾는 방법으로서 사용되어 왔다. 유전자 알고리즘은 종래의 검색 알고리즘과 달리 검색 공간 중 하나의 점뿐만 아니라 많은 점을 동시에 고려하기 때문에 국부최소점(local minima)점에 수렴되는 현상을 극복할 수 있다.

유전자 알고리즘은 모든 해들이 유전자 구조로 만들고, 입력값 집단을 랜덤하게 선택한후 이를 보다 최적 해에 가까운 집단으로 진화시키는 세대교체(generations)과정을 반복수행한다.[12]

2.1 초기생물집단의 발생

유전자 알고리즘에서는 복수집단의 경쟁을 통해 문제를 해결한다. 그러므로 유전자 알고리즘을 개시하는 시점에서는 그 집단은 설정되어 있지 않기 때문에, 이 과정에서 초기 집단을 발생시킨다. 일반적인 집단 발생에 있어서 어떠한 개체가 좋은가를 모르기 때문에 이 초기집단은 랜덤하게 정하여진다.

2.2 각 개체의 적응도의 계산

생물집단중의 각 개체에 대해 적응도 계산을 행한다. 적응도 평가는 유전자 알고리즘과 최적화 문제간의 인터페이스 역할을 수행한다. 유전자 알고리즘은 구조에 대해서 직접정보를 사용하는 것이 아니고, 각 개체에서 생성된 정보에 따라 해(solution)에 대한 각각 개체품질을 사정한다. 제안된 해결안의 품질은 일반적으로 그 해결안이 요구된 작용을 어느 정도 실행했는가, 그래서 주어진 제약을 어느 정도 만족했는가에 의해 계산된다.

2.3 집단평가

생성된 집단이 유전자 알고리즘의 과정을 수료시키는 조건을 만족하고 있는가를 조사하여, 만족하고 있다면 과정을 종료시킨다. 여기서 종료조건은 세대교대(새로운 생물집단이 생성되는 것)의 회수가 정해진 회수에 달하는 것을 의미한다.

2.4 유전자의 교차(crossover)

먼저 생성된 생물 집단 중 교차시킬 개체를 선택해야 한다. 일반적으로 토너먼트 선택법을 이용한다. 이 방법은 우선 생물집단중의 2쌍의 개체를 랜덤하게 선택하여 각 쌍에서 적응도가 높은 개체를 선택해 낸다.

교차방법은 우선 랜덤하게 분할유전자(split gene)를 결정한다. 이것을 중심으로 개체 1, 개체 2의 유전자를 교체시킨다. 이것에 의해 새로운 개체의 생성이 가능하다. 또한 분할 유전자는 1점 교차법으로 교차되고, 이것은 랜덤하게 결정되어진 교차위치에서 전달한 부분적인 유전자형을 교체하는 것에 의해 새로운 유전자를 만드는 것이다.

발생한 새로운 유전자를 사용하여 과정 2)부터 4)까지 일정한 횟수 동안 반복하여 최적의 해를 탐색한다.

3. 로봇의 전역경로 탐색

본 논문에서는 유전자 알고리즘을 이용한 자율이동로봇의 전역 이동경로 계획방법의 그림 1과 같다. 먼저 로봇이 이동하여야 할 전체 영역에 대한 지도가 주어지고 주어진 지도를 셀 분해법에 의해 셀 영역으로 분해한다. 분해된 셀은 로봇과 장애물을 점유하는 지도로 재구성되며, 장애물의 모서리영역을 탐색한다. 모서리영역들은 그래프의 형태를 가진 자료구조로서 재구성되어지며, 모서리영역을 이동하는 최단 경로를 유전자 알고리즘을 통하여 탐색하게 된다

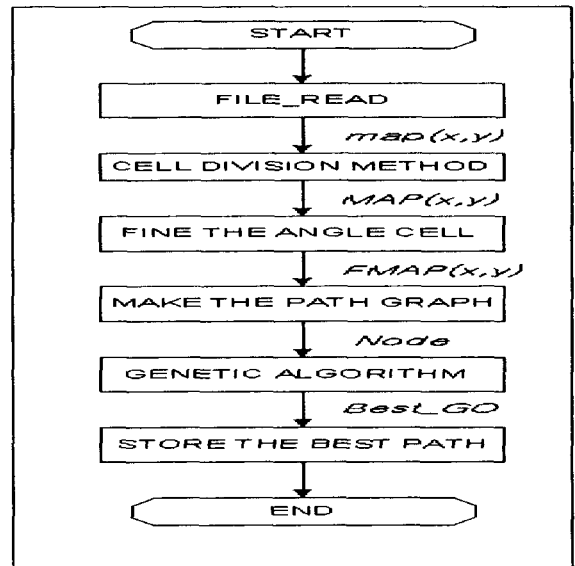


그림 1. 유전자 알고리즘을 이용한 전역경로 계획방법 순서도
 Fig. 1. The flowchart of global path planning using genetic algorithm

3.1 셀 분해법

비전시스템 또는 미리 저장된 데이터로부터 지도를 얻는다. 지도의 형태는 식 1과 같다. 장애물이 있는 픽셀은 1로 그렇지 않은 경우는 0으로 설정되어 있다.

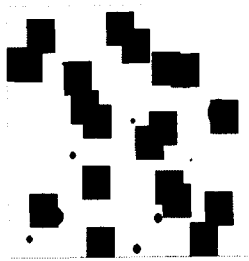
$$map(x, y) = \begin{cases} 1: obstacle \\ 0: otherwise \end{cases} \quad (1)$$

로봇이 이동하면서 매 샘플링 시간마다 온라인으로 경로계획을 수행하기 위해서는 빠르고 반복적인 알고리즘이 필요하다. 셀 분해법은 활동영역을 셀 단위로 처리하기 때문에 활동 영역에 대한 데이터가 줄어들기 때문에 과정이 간단하고 반복적으로 구현된다는 장점이 있다.

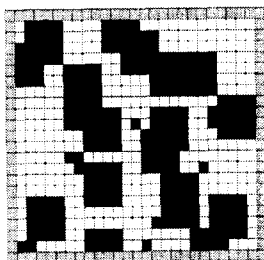
셀 분해법은 일정 셀 단위로 지도에서 장애물을 분해한다. 지도로부터 입력된 집합 map 을 셀 단위로 분해된 집합 MAP 으로 변환한다.

$$C_{DIV} : map \rightarrow MAP \quad (2)$$

장애물의 곡면을 자세히 묘사하기 위해 작은 셀로 나누는 경우 정보량이 오히려 많아지기 때문에 여기서는 한 셀을 로봇이 통과할 수 있는 적당한 크기로 나누고, 한 셀의 면적의 10% 이상을 장애물이 차지 할 경우는 장애물이 존재하는 셀로 설정하도록 하였다.



(a) 집합 map



(b) 집합 MAP

그림 2 셀분해법

Fig. 2 The cell division method

$$MAP = \{M: M(x, y), x=0 \sim W_{MAP}-1, y=0 \sim H_{MAP}-1\} \quad (3)$$

$$W_{MAP} = \frac{W_{map}}{CWH}, \quad H_{MAP} = \frac{H_{map}}{CWH}$$

$$CWH = \max(W_R, H_R)$$

$$M(x, y) = \begin{cases} 0 : O_{PA} < 10 \\ 1 : x=0 \text{ or } y=0 \text{ or } x=W_{MAP}-1 \\ \quad \text{or } y=H_{MAP}-1 \\ 2 : O_{PA} \geq 10 \end{cases}$$

$$O_{PA} = \frac{A_{OB}}{A_{CELL}} \times 100,$$

(4)

$$A_{OB} = \sum_{x=0}^{CWH-1} \sum_{y=0}^{CWH-1} map(x * W_{MAP} + X, y * H_{MAP} + Y)$$

$$A_{CELL} = CWH^2$$

W_{MAP}, H_{MAP} 은 셀분해된 지도 MAP 의 행과 열의 갯수를 의미하며, W_{map}, H_{map} 은 각각 제공된 지도 픽셀의 행과 열의 갯수를 의미한다. W_R 과 H_R 은 로봇의 폭과 높이를 의미하며, CWH 는 로봇의 폭 또는 높이 중 큰 값을 의미한다. A_{OB} 는 장애물의 면적이고, A_{CELL} 은 셀의 넓이이다.

3.2 장애물 모서리 셀 탐색

본 연구에서는 로봇의 이동 경로의 경우 수를 감소시키기 위하여 셀 분해로부터 얻은 지도에 가시도표(visibility graph)방법을 적용하여 장애물의 모서리를 탐색하였다.

장애물 모서리 탐색알고리즘은 최단 경로 정의 집합의 수를 줄일 수 있는 방법의 하나로서 만약 장애물이 최단 경로 상에 놓여 있을 경우는 장애물의 모서리를 따라 움직이는 것이 최단거리 임을 이용한 것이다. 이 방법을 적용함으로써 장애물이 없는 경우나 적을 경우 불필요한 최단경로 탐색 시간을 줄일 수 있다. 셀분해된 MAP 의 집합을 이용하여 새로운 $FMAP$ 를 생성한다.

$$F_{EDGE} : MAP \rightarrow FMAP \quad (5)$$

$$FMAP = \{FM: FM(x, y), x=0 \sim W_{MAP}, y=0 \sim H_{MAP}\} \quad (6)$$

$$FM(x, y) = \begin{cases} 3 : FMC = TRUE \\ M(x, y) \text{ Otherwise} \end{cases}$$

$$FMC = \{ M(x, y)=0 \text{ and } \{ \begin{aligned} &\{M(x-1, y-1)=2 \text{ and } M(x, y-1)=0 \\ &\quad \text{and } M(x-1, y)=0\} \\ \text{or } &\{M(x-1, y-1)=2 \text{ and } M(x, y-1)=0 \\ &\quad \text{and } M(x-1, y)=0\} \\ \text{or } &\{M(x-1, y-1)=2 \text{ and } M(x, y-1)=0 \\ &\quad \text{and } M(x-1, y)=0\} \\ \text{or } &\{M(x-1, y-1)=2 \text{ and } M(x, y-1)=0 \\ &\quad \text{and } M(x-1, y)=0\} \} \} \end{aligned} \quad (7)$$

$$x > 0, y > 0$$

$FMAP$ 은 MAP 에서 얻어진 $M(x, y)$ 에서 모서리의 셀을 찾아 $FM(x, y)$ 의 값을 3으로만 치환한다. 장애물 모서리 탐색을 통하여 장애물을 회피하여 이동할 수 있는 지역 최단 경로 정보를 얻을 수 있었다. 지금까지 과정으로 얻어진 $FMAP$ 은 그림 3과 같으며 $FM(x, y)$ 의 값은 상수 인 0(자유주행영역), 1(지도 가장자리), 2(고정장애물), 3(장애물 모서리) 등으로 설정됨을 볼 수 있다.

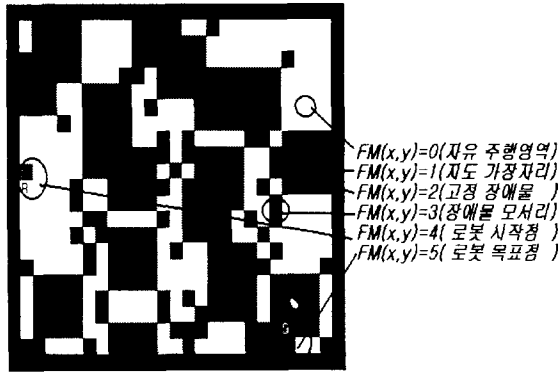


그림 3 그림 2의 MAP 대한 집합 FMAP
Fig. 3 The FMAP of the MAP in Fig. 2

다음 과정으로서 얻어진 모서리 노드를 연결하여 임의의 경로를 설정하기 위한 그래프 자료구조 형식을 가진 NODE 정보군을 생성한다. 본 논문에서는 집합 NODE를 생성하기 위해서 각 모서리에서 연결 가능한 다른 모서리 노드를 탐색한다. 모서리 노드 N_i 의 경로 그래프의 자료구조는 다음과 같은 구조체로서 정의가 되어 있다.

$$NODE = \{N : N_i = \{x, y, Nnode, NC\}, i : 0 \sim Tnode, FM(x_{N_i}, y_{N_i}) = 3\} \quad (8)$$

$$NC_{N_i} = \{NC_0, NC_1, \dots, NC_n\}, n = Nnode - 1$$

x, y 는 셀분해된 좌표계에서 모서리 노드 N_i 의 위치이며, $Nnode$ 는 모서리 노드 N_i 에서 연결 가능한 노드번호 집합인 NC 의 원소 갯수이다. $Tnode$ 는 전체 장애물 모서리 노드 갯수이다.

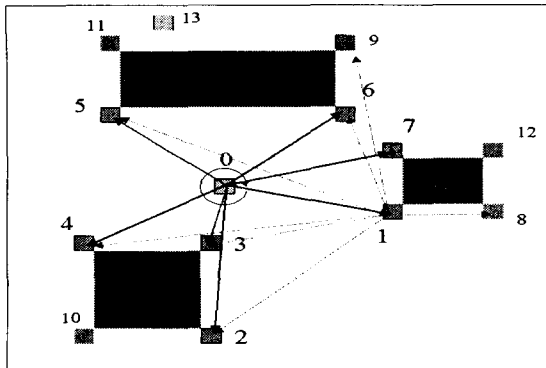


그림 4. 모서리 노드 정보군
Fig. 4. Angle node informations.

그림 4에는 모서리 노드 정보군에서 연결그래프를 생성하는 과정을 볼 수 있다. 0번 노드는 로봇의 시작 위치이며, 0번 노드에서부터 탐색을 하여 전체 노드까지 확장 탐색한다. 그림 4에서 0번 노드에 대한 탐색 내용을 실선으로, 1번 노드에서의 탐색을 점선으로 표현하였다. 그림 4의 내용에서 얻어진 결과는 전체 노드갯수 $Tnode$ 는 14이며, 목표점 노드는 13이다. 0번 노드에 대한 구조체 집합은 $N_0 = \{100, 100, 7, \{1, 2, 3,$

4, 5, 6, 7\}\}이며 1번 노드에 대한 구조체 집합 $N_1 = \{150, 120, 7, \{8, 2, 3, 4, 5, 6, 9, 7\}\}$ 이다. 같은 방법으로 전체 노드에 대한 연결 그래프를 형성시킬 수 있다. 다음 과정에서는 얻어진 연결 그래프를 이용하여 최단 경로를 탐색하는 과정을 수행한다.

3.3 유전자알고리즘 적용

본 논문에서 사용한 유전자알고리즘의 순서도는 그림 5와 같다. 초기의 모집단을 생성한다. 생성된 모집단에 대한 평가를 하여 생성된 4개의 경로집합을 적응도가 높은 순서로 정렬한다.

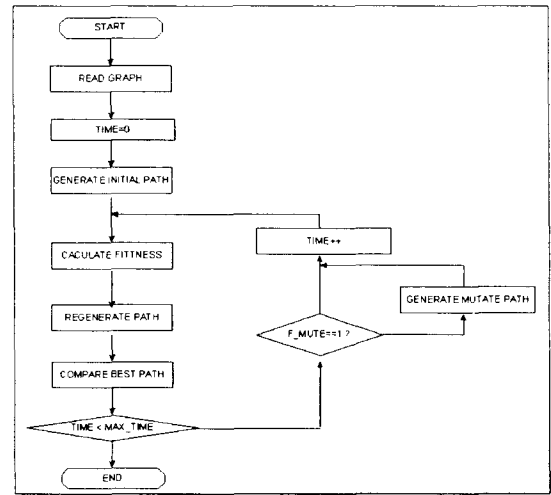


그림 5 유전자 알고리즘 적용 순서도
Fig. 5 The flowchart for adapting the genetic algorithm

다음세대를 위하여 두 가지의 과정을 수행한다. 첫 번째 과정은 교차이고, 두번째 과정은 돌연변이이다. 먼저 교차과정에서는 기존의 최 우수집단과 4개의 생성된 집단 중 첫번째 집단을 교차시켜 다음세대 집단 0,1을 만들고, 두번째 집단과 교차시켜 다음세대 집단 2,3을 생성한다. 다음으로 돌연변이의 과정을 보면 1/3확률값으로 랜덤하게 발생하는 돌연변이 플래그에 의해 동작되며, 방법으로는 다음 기존의 최우수집단과 4개의 우수집단 중 첫번째 집단에서 임의의 점을 택하고, 그 점부터 랜덤하게 경로를 설정하여 새로운 집단을 만든다. 또한 생성된 최 우수집단은 기존의 최 우수집단과 비교하여 적응도를 비교하여 최 우수집단을 갱신한다. 생성된 집단은 재평가를 통하여 우수집단을 선택 및 교차하는 과정을 반복한다. 일정 횟수를 반복하여 가장 좋은 경로를 최단 경로로 선택한다.

3.3.1 초기 모집단 생성

유전자알고리즘을 적용하기 위하여 먼저 모서리 경로탐색에 의해 얻어진 가능경로의 그래프를 읽는다. 입력된 가능경로를 바탕으로 로봇의 위치를 초기 노드로 하여 랜덤하게 초기의 모집단을 설정한다. 식 9를 사용하여 초기 모집단을 얻는다. 초기의 모집단 갯수(k)는 4로 설정하였다.

$$GO = \{G; G_i^k = \begin{cases} 0 & i=0 \\ NC_{l(N_m)} & 1 < i < NGO^k \\ \text{random}(Npath_{G_{i-1}^k}) & i = NGO^k + 1 \end{cases} \quad (9)$$

$, l = \text{random}(Npath_{G_{i-1}^k}), m = G_{i-1}^k, k = 0 \sim 3$

$NC_{l(N_m)}$: 노드 N_m 의 NC 집합중 l 번째 원소
 NGO^k : k 번째 경로집합의 총 경로노드수
 G_i^k : k 번째 경로집합에서 i 번째 경로노드
 $Npath_{G_{i-1}^k}$: k 번째 경로집합에서 $i-1$ 번째 경로노드의 이동가능 노드수

GO 집합은 $NODE$ 집합에서 원소들을 랜덤하게 선택하는 것과 같다. 즉 노드 N_i 의 $Npath$ 사이의 값 중에 랜덤한 수 l 를 생성하여 N_i 와 연결 가능한 노드에 대한 정보집합(NC)에서의 l 번째 노드번호를 참조하여 G^k 집합의 i 번째 원소로 설정한다. 다음 G^k 집합의 $i+1$ 번째 원소는 G^k 집합의 i 번째 원소에 대한 노드 정보를 참조하여 설정한다. 두 노드간 무한 루프에 빠지는 것을 방지하기 위하여 p 노드에서 q 노드로 이동하였을 경우에 p 와 q 의 연결을 끊는다. 즉 p 노드에서 이동가능 노드번호에서 q 번을 삭제하고, q 노드에서 이동가능 노드번호 중에서 p 번을 삭제한다.

G^k 의 원소인 노드번호가 목표점의 노드번호와 같거나 $Npath_{G_{i-1}^k}$ 의 값이 0일 경우에는 G^k 집합에 대한 경로설정을 종료한다.

3.3.2 평가

언어진 경로들에 대한 평가를 실시한다. 본 논문에서 사용한 평가 함수는 식 10, 식 11과 같다. 평가함수 fit 는 거리함수에 반비례하며, 최종 노드 값이 목표점의 노드 값인지를 판별하여 아닐 경우는 거리함수에 가중치 $NGBEST$ 를 추가하였다. $NGBEST$ 는 목표점에 도달하지 않은 경로중 가장 많은 노드를 이동한 경로의 이동노드 수이다. 적합도 평가는 경로들의 거리함계로서 계산되며 적합도가 높은 것이 좋은 경로이다. 또한 목표점까지 경로가 설정되지 못한 경우는 가장 많은 경로를 거친 경로가 좋은 경로이다.

$$Dist^k = \begin{cases} \sum_{i=1}^{NGO^k} \sqrt{(x_{N_i} - x_{N_{i+1}})^2 + (y_{N_i} - y_{N_{i+1}})^2} \\ \text{, if } (x_{N_{NGO^k}} = x_{Goal} \text{ and } y_{N_{NGO^k}} = y_{Goal}) \\ \sum_{i=1}^{NGO^k} \sqrt{(x_{N_i} - x_{N_{i+1}})^2 + (y_{N_i} - y_{N_{i+1}})^2} \\ \text{+ } (NGBEST - NGO^k) * K_p \\ \text{, if } (x_{N_{NGO^k}} \neq x_{Goal} \text{ and } y_{N_{NGO^k}} \neq y_{Goal}) \end{cases} \quad (10)$$

K_p : 비례상수 (=25(본 논문에서 임의로 설정한 상수값))

$$fit^k = \frac{100}{Dist^k} \quad (11)$$

3.3.3 저장 및 진화

평가함수의 값을 키 값으로 하여 내림차순으로 정렬한다. 가장 fit 값이 높은 경로가 SG^0 에 위치하게 되고, 그 다음으로

SG^1, SG^2, SG^3 으로 위치하게 된다.

$$SGO = \text{sort}(GO) \quad (12)$$

SG^0 와 $BEST_GO$ 의 적용도를 비교하여 좋은 것을 $BEST_GO$ 로 저장, 갱신한다.

$$\begin{aligned} &\text{if } fit_{SG^0} > fit_{BEST_GO} \text{ then} \\ &\text{begin} \\ &\quad exBEST_GO = BEST_GO \\ &\quad BEST_GO = \text{copy}(SG^0) \\ &\text{end} \end{aligned} \quad (13)$$

3.3.4 선택, 교차, 돌연변이

본 논문에서는 새로운 세대에 대한 경로를 위하여 SG^0 경로와 SG^1 의 경로를 선택하고, $BEST_GO$ 와 교차 또는 돌연변이를 실시한다.

$$\begin{cases} \text{cross}(SG^0, exBEST_GO) & : CrossFlag = 0 \\ \text{cross}(SG^0, SG^1) & : CrossFlag = 1 \\ \text{RanGO}(SG^0, SG^1) & : CrossFlag = 2 \\ \text{RanGO}(SG^2, SG^3) & : CrossFlag = 3 \end{cases} \quad (14)$$

교차는 $CrossFlag$ 에 의해 작업이 분리된다. $CrossFlag$ 0, 1은 교차를 수행하여 새로운 경로집합을 얻을 경우에 설정되고, 2, 3은 교차 실패 또는 돌연변이를 수행할 경우 새로운 경로집합을 얻고자할 때 설정된다. $\text{cross}(A, B)$ 함수는 경로 A 와 경로 B 에서 임의의 공통 노드를 찾아 경로를 교차하여 새로운 경로를 생성하는 함수이며, $\text{RanGO}(A, B)$ 는 경로 A, B 에서 임의의 노드로부터 이동경로계획을 다시 수행하여 새로운 경로를 얻는 함수이다.

식 15에 $\text{cross}(SG^1, SG^2)$ 함수에 대한 예를 나타내었다. 랜덤한 수 2를 얻었을 경우 SG^1 는 1이므로 SG^2 에서 1을 탐색한다. SG^1 에서 1이하의 노드 즉 {2,3,4,3,6}과 SG^2 에서 4이하의 노드 {4,8,9,6}를 서로 비교하여 다음으로 같은 노드를 탐색한다. 노드 4가 같이 존재하고 있으므로 새로운 경로를 위하여 SG^1 에서는 {1,2,3,4}와 SG^2 {1,4}가 서로 교차되어 새로운 경로 $NewG^1$ 과 $NewG^2$ 를 얻었다.

$$\begin{aligned} SG^1 &= \{0, 1, 2, 3, 4, 3, 6\} \\ SG^2 &= \{0, 1, 4, 8, 9, 6\} \\ \text{cross}(SG^1, SG^2) \\ NewG^1 &= \{0, 1, 4, 3, 6\} \\ NewG^2 &= \{0, 1, 2, 3, 4, 8, 9, 6\} \end{aligned} \quad (15)$$

만약 SG^1 과 SG^2 의 교차를 위한 랜덤 $Node$ 를 찾지 못했을 경우와 돌연변이 프래그가 1일 경우에는 $\text{RandGo}()$ 함수를 수행한다. $\text{RandGo}(A, B)$ 함수는 랜덤한 수 $N1$ 을 발생시키고 A 와 B 의 경로를 $N1$ 이후부터 초기 모집단 설정과 같은 방법으로 계속 진행시킨다. 식 16에 RandGo 에 대한 함수의 예를 볼 수 있다. SG^1 과 SG^2 에 대한 같은 노드를 0번 노드 이외에는 찾을 수 없다. 그러므로 $\text{RandGo}()$ 함수를 실행한다. 랜덤한 수를 3을 선택하였을 경우 SG^1 에서는 9번 노드에서 10번이 아닌 다른 로드 11번을 선택하여 경로 설정 과정

을 수행한다. SG^2 의 경우도 3번째 노드 2이하부터 랜덤 경로 설정을 진행한다.

$$\begin{aligned}
 SG^1 &= \{0, 8, 9, 10\} \\
 SG^2 &= \{0, 1, 2, 3, 4, 3, 5, 7, 6\} \\
 RandGo(SG^1, SG^2) & \\
 NewG^1 &= \{0, 8, 9, 11, 6\} \\
 NewG^2 &= \{0, 1, 2, 5, 9, 6\}
 \end{aligned}
 \tag{16}$$

다음 세대로 얻어진 $NewG^k$ 의 경로를 다음 세대의 G^k 로 이동시키고 $max_generation$ 까지 평가, 선택, 교차, 돌연변이 과정을 반복수행하여 최적의 경로를 도출해 나간다.

$$REGENERATE : G^k = NewG^k \tag{17}$$

4. 모의실험

본 논문에서 제안한 유전자 알고리즘을 이용한 최단 경로 탐색방법을 여러가지 지도에 적용하여 그 효과를 검증하였다. 모의실험의 과정은 화상자료를 BMP 파일 형식으로 저장하고, 저장된 지도 내용을 읽어들이 셀 분해법을 이용한 새로운 셀 단위의 지도를 새로이 얻는다. 얻어진 지도에서 장애물의 사각 모서리 정보를 얻고 가능 경로 그래프를 도출하였다. 그래프를 랜덤하게 발생하여 초기 모집단을 설정하고 평가, 선택 및 교차, 돌연변이의 과정을 반복하였다. 전체 모의 실험은 5가지 장애물들에 대하여 수행하였다. 1차 모의실험은 비교적 장애물이 많지 않은 경우에 대하여 모의실험 하였으며, 2차 모의실험은 기존연구와 비교하기 위해 비슷한 장애물 환경에서 수행하였다. 3차, 4차와 5차는 각각 오목 장애물, 복잡한 장애물 및 좁은 통로에서의 유전자 알고리즘 성능 평가를 수행하였다.

1차 모의실험 지도로서 그림 6과 같은 장애물을 입력하였다. $CWH=15pixel$ 로 설정하였으며, $N_R=N_0=\{2,2,2,\{1,2\}\}$ 로 설정하였다. 전체 노드 갯수($Tnode$)는 10을 얻었으며, 각각의 노드에 대한 번호는 그림 6에서 각각 노드에 위에 적힌 번호와 같다.

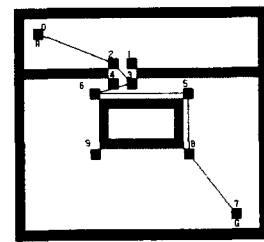
$$Node = \{N_0, N_1, \dots, N_{Tnode-1}\} \tag{18}$$

그림 6의 (a)는 초기 모집단에서의 최적경로를 표현한 것이며, $BEST_GO = \{0, 2, 3, 6, 5, 8, 7\}$ 을 얻었다. 그림 6의 (b)는 17번 진화 후, 즉 $generation=17$ 일 경우의 $BEST_GO$ 경로를 사선으로 표현한 것이다. 여기서 $BEST_GO = \{0, 2, 3, 5, 8, 7\}$ 을 얻었다. 각각에 대한 fit 값은 다음 표 1과 같이 얻었다. $max_generation$ 은 100[generation]으로 설정하였으며, $max_generation$ 까지 반복을 하였을 경우에도 그림 6의 (b) 경로를 계속 유지하였다. 같은 장애물을 15번 반복을 실험을 하였으며, 그 결과 평균 최단 경로의 수렴 정도는 10.1[generation]이었다.

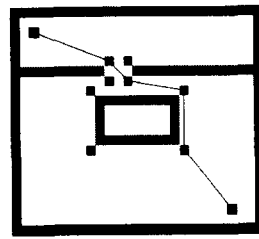
표 1 모의 실험 1의 거리와 평가 결과 값

Table 1 Distance and Fitness Results of Experimental Simulation 1

generation	1	17
$Dist_{BEST_GO}$	31.606	30.455
Fit_{BEST_GO}	3.16	3.28



(a)



(b)

그림 6. 모의 실험 1 결과

Fig 6. The result of experimental simulation 1

2차 모의실험은 기존연구와 비교하기 위해 비슷한 장애물 환경에서 모의실험을 수행하였다. 표 2는 2차 모의실험에 대한 거리함수 값과 적합도 값을 나타내었다. 모의 실험 결과 기존 연구에서는 약 23[generation] 반복 진화를 통해 최단 거리를 탐색하였으나 본 연구에서는 약 12[generation]에 최단거리를 탐색하였다. 모의실험은 50[generation] 정도 실시하였으며, 실험결과 안정적으로 최단경로를 설정함을 확인하였다. 그림 7에 (a)는 초기모집단, (b)는 2[generation] 진화, (c)는 9[generation] 진화된 경로를 나타낸 것이다.

표 2 모의 실험 2의 거리와 평가 결과 값

Table 2 Distance and Fitness Results of Experimental Simulation 2

generation	1	2	9
$Dist_{BEST_GO}$	44.861	32.58	25.589
Fit_{BEST_GO}	2.23	3.07	3.91

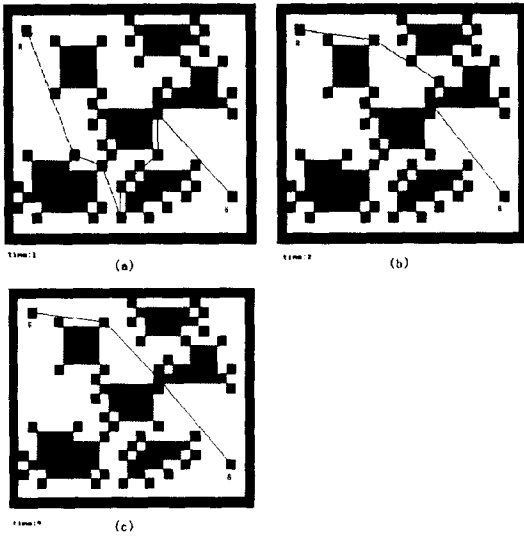


그림 7. 모의 실험 2 결과

Fig. 7. The result of experimental simulation 2

3차, 4차와 5차 모의실험은 기존연구에서 제시치 못한 오목, 복잡한 지도 및 좁은 통로에 대한 경로설정과정에 대한 모의실험을 수행하였다. 그림 8, 9, 10과 표 3, 4, 5는 각각 3차, 4차 그리고 5차 모의실험에 대한 결과 및 거리함수 값과 적합도 값을 나타내었다. 모의 실험 결과 기존 연구에서는 최단 경로를 탐색하지 못했으나 본 연구는 안정적으로 최단 거리를 탐색하였다. 특히 모의 실험 4를 위한 지도는 그림 2의 (a)와 같고 그림 2의 (b)와 같이 셀분해법에 의해 $FMAP$ 을 얻었다. 주어진 $FMAP$ 에 장애물 모서리 탐색을 하고, $NODE$ 를 그림 3과 같이 얻었다. $Tnode$ 는 38이었으며 초기 시작점인 로봇 노드는 $N_R = N_0 = \{1, 10, 8, (1, 2, 3, 4, 5, 6, 7, 8)\}$ 과 같이 얻었으며, 목표점 노드는 $N_G = N_{37} = \{22, 21, 1, (36)\}$ 과 같이 설정하였다. 먼저 초기의 모집단 중 $BEST_GO$ 집합으로 그림 9(a)와 같이 얻었다. 그림 9(b),(c),(d)는 각각 $generation=9$, $generation=24$, $generation=67$ 일 경우에 얻은 $BEST_GO$ 집합을 실선으로 표현한 것이다. 표 4에 모의실험 4의 fit 값의 변화를 표현하였다. 그림 9(a),(b)는 목표점에 도달하지 않은 경로들이 최적 경로로서 얻어진 경우이다. 이 경우 적합도 함수 값에는 $NGBEST$ 가 적용되어 있으므로 많은 모서리 노드를 경유한 경로가 최적경로로서 설정되었다. $generation=67$ 이후에는 계속적으로 $BEST_GO$ 가 유지되어 바뀌지 않음을 볼 수 있었다. $max_generation$ 을 100으로 모의실험 1과 같이 설정하였다. 10번의 실험을 반복 수행하였으며 평균 80번에 수렴함을 보였다. 모의 실험 1에서와 거의 비슷한 100번 이하의 탐색으로도 최단 경로를 설정할 수 있음을 모의실험을 통하여 알 수 있었다.

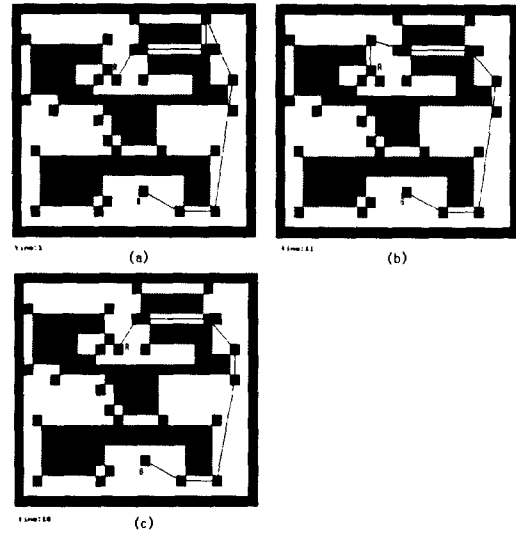


그림 8. 모의 실험 3 결과

Fig. 8. The result of experimental simulation 3

표 3 모의 실험 3의 거리와 평가 결과 값

Table 3 Distance and Fitness Results of Experimental Simulation 3

generation	1	11	17
$Dist_{BEST_GO}$	42.939	41.405	37.518
Fit_{BEST_GO}	0.43	2.42	2.67

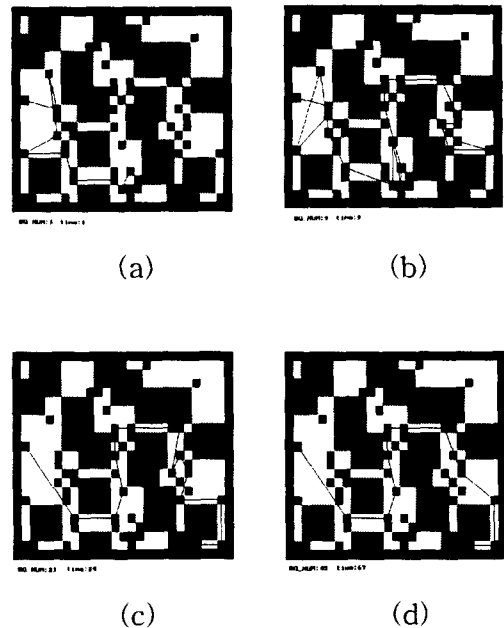


그림 9. 모의 실험 4의 결과

Fig. 9. The result of experimental simulation 4

표 4 모의 실험 4의 거리와 평가 결과 값
Table 4 Distance and Fitness Results of Experimental Simulation 4

generation	1	9	24	67
Dist _{BEST_GO}	2035.539	1787.271	59.019	50.671
Fit _{BEST_GO}	0.05	0.06	1.69	1.97

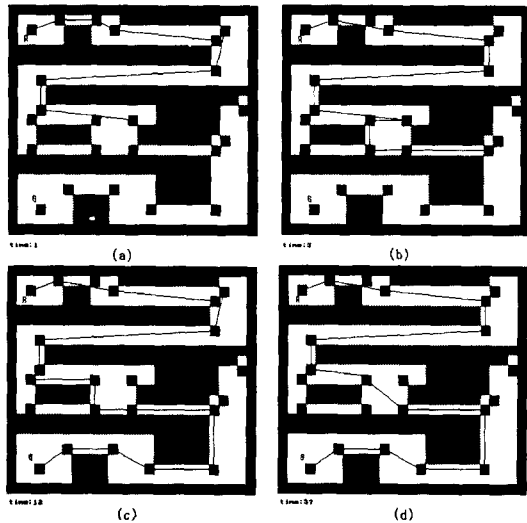


그림 10 모의 실험 5의 결과
Fig 10 The result of experimental simulation 5

표 5 모의 실험 5의 거리와 평가 결과 값
Table 5 Distance and Fitness Results of Experimental Simulation 5

generation	1	5	12	57
Dist _{BEST_GO}	2058.057	2026.781	93.517	90.346
Fit _{BEST_GO}	0.05	0.06	1.07	1.11

5. 결 론

본 논문에서는 유전자 알고리즘을 이용한 이동로봇의 전역 경로 설계방법을 제안하였으며, 제시한 방법은 유전자알고리즘을 적용하기에 적합하도록 지도를 재맵핑하는 전처리과정을 수행하고 유전자알고리즘을 적용한 경로를 생성, 진화시키면서 최적경로를 탐색을 수행하는 것이다. 유전자 알고리즘을 사용함으로써 목표점의 변동과는 무관하게 최단의 전역 경로에 안정적 수렴을 하는 것을 확인하였으며, 오목 다면체나 원형 또는 복잡한 통로의 최적 경로 탐색 또한 안정적으로 빠른 탐색을 나타냄을 확인하였다.

전역경로 계획과정은 먼저 지도를 입력받고, 이것을 셀단위로 분해하여, 모서리 탐색에 의한 가시 그래프를 생성하는 과정을 거친다. 여기에서 얻어진 가시그래프의 정보로부터 최적의 전역경로를 탐색하기 위해 유전자 알고리즘 적용하였다. 유전자 알고리즘에서 초기의 랜덤한 경로 모집단에서 출

발하여 가장 *fit*값이 높은 경로들만을 선택하여 진화시키면서 최적의 경로를 찾는다. 교차방법은 가시그래프로부터 얻어진 *node*들로 구성된 경로에서 2개의 교차점을 찾고 *BEST_GO*와 *SG⁰*, *SG¹*에서 교차점 사이의 값들을 서로 교차하여 경로를 진화시킨다. 돌연변이방법은 *SG⁰*와 *BEST_GO* 경로에서 임의의 점을 결정 후 다른 경로를 생성할 수 있도록 *node* 정보를 수정하여 랜덤 경로를 설정한다.

제한한 방법을 검증하기위한 모의실험에서는 전체 27×23 셀로 구성된 지도에서 장애물의 갯수와 관계없이 약 100회 이내의 반복 탐색으로 최단 경로가 설정됨을 확인하였다. 본 논문에서 제안한 전역경로 설정방법은 지도상에 존재하는 경의 수가 기하급수적임에도 불구하고, 또한 수식적인 전개 없이 유전적 적자생존의 원리를 이용하여 효과적으로 최단 경로가 설정됨을 확인하였다.

본 연구에서 얻어진 전역경로를 기준 이동 경로로 사용하여 향후 여러 알고리즘을 적용한 지역경로계획에 의해 장애물을 효율적으로 회피하면서 전역경로를 추종할 수 있는 알고리즘을 설계할 계획이며 3차원 이상의 다차원 영역에서도 본 논문의 알고리즘이 적용될 수 있도록 확장 연구코자 한다.

이 논문은 1998년도 조선대학교 학술연구비의 지원을 받아 연구되었음.

참고 문헌

- 1) Phillip John McKerrow, "Robotics", Addison-Wesley Pub., Co., 1993.
- 2) Lozano-Perez, T. and Wesley, M. A., "An Algorithm for Planning Collision-Free Path Among Polyhedral Obstacles", Comm. of ACM, Vol. 22, NO. 10. October 1979.
- 3) S.K. Ghosh and D.M. Mount, 1991. "An Output-Sensitive Algorithm for Computing Visibility Graphs", SIAM J. Comput., 20, 888-910. 1991
- 4) Michael T. Goodrich, Colm O'Dunlaing, Chee K. Yap, "Constructing the Voronoi Diagram of a Set of Line Segments in Parallel", Algorithmica 9(2), 128-141, 1993.
- 5) Jarvis, R.A. and Byrne J.C., "Robot Navigation: Touching, Seeing, and Knowing", Proceedings of the First Intern. Conf AI, Nov. 18-21, 1986.
- 6) Latombe, J., "Robot Motion Planning", Norwell, Ma: Kluwer Academic Publishers, 1991
- 7) Khosla and R. Volpe, "Superquadric artificial potentials for obstacle avoidance and approach", IEEE conf. on Robotics and Automat., 1778~1784, 1988.
- 9) 윤창경. "유전알고리즘을 이용한 이동 로봇의 충돌회피

경로 계획에 관한 연구”, 한국과학기술원 석사학위 논문, 1993.

- 10) I. Ashiru and C. A. Czarnecki, "Optimal motion planning for mobile robots using genetic algorithms.", IEEE International Conference on Industrial Automation and Control, Hyderabad, January 1995.
- 11) David E. Goldberg., "Genetic Algorithms in Search", Optimazation and Machine Learning. Addison-Wesley, 1989.

- 12) Alan C. Schultz, "Learning Robot Behaviors Using Genetic Algorithms". Proceedings of the First World Automation Congress, 607-612, TSI Press: Albuquerque, NM. 1994

저 자 소 개



최 한 수 (崔 漢 秀)

1954년 9월 30일생. 1980년 조선대학교 전기공학과 졸업. 1994년 전북대학교 대학원 졸업(공학박사). 1985년~1986년 미국 Oregon Graduate Center 객원교수. 현재 조선대학교 공과대학 전기·제어계측공학부 교수

Tel : 062-230-7185

E-mail : hschoi@mail.chosun.ac.kr



정 헌 (鄭 憲)

1970년 10월 13일생. 1993년 조선대학교 제어계측공학과 졸업. 1996년 동 대학원 제어계측공학과 졸업(석사). 1999년 동 대학원 제어계측공학과 졸업(박사).1992년~1995년 (주)금호 사원. 1996년~1997년

한국기계연구원 위촉연구원. 1998년~현재 조선대학교 에너지자원신기술연구소 전임 연구원

Tel : 062-230-7185

E-mail : j_h@shinbiro.com