

유전 알고리즘을 이용한 자율 이동로봇의 최적경로 계획

Path planning of Autonomous Mobile robot based on a Genetic Algorithm

°이 동 하*, 이 만 형**

* 부산대학교 대학원 (Tel: 051-510-1456; Fax: 051-512-9835; E-mail:ldh0904@mecha.re.pusan.ac.kr)

** 부산대학교 기계공학과 (Tel: 051-510-2331; Fax: 051-512-9835; E-mail:mahlee@hyowon.pusan.ac.kr)

Abstract

In this paper we propose a Genetic Algorithm for the path planning of an autonomous mobile robot. Genetic Algorithms(GAs) have advantages of the adaptivity such as GAs work even if an environment is time-varying or unknown. Therefore, we propose the path planning algorithms using the GAs-based approach and show more adaptive and optimal performance by simulation.

1. 서 론

최근 효율적인 작업을 하기 위하여 자동화에 관한 관심이 더욱 커지고 무인 반송 시스템의 구현을 위한 AGV(Automatic Guided Vehicle)과 자율 이동로봇(AMR) 등에 관한 연구도 활발히 진행되고 있다.

이러한 반송 로봇들은 일정한 작업환경 하에서 원활한 작업을 수행하기 위하여 자율적으로 환경을 인식하고 변화하는 상황에 알맞은 동작이 가능하여야 한다. 이를 구현하기 위하여 원하는 목표 지점까지 충돌 없이 자유롭게 이동할 수 있게 하고 효율적인 작업 수행을 위해 목표점까지의 최단 경로를 통한 이동이 가능하게 하기 위한 최단 경로계획 알고리즘에 관하여 연구에 많이 되어왔다.

이러한 최적 경로계획 알고리즘에는 여러 가

지 방법이 있다. 첫 번째로 미적분학을 기본으로 한 방법은 미분과 적분이 가능하여야 함으로 이동로봇이 작업할 환경을 수학적으로 모델링 해야 하는 것이 필요하다. 하지만 다양한 환경 모델링한다는 것은 쉬운 문제는 아니며, 국소점이 존재할 경우 이를 벗어날 수 있는 방법을 제공하지 않아 국소점에 빠질 우려가 있다. 두 번째로 복수 계산을 기본으로 하는 방법은 가능한 모든 경우의 수를 계산하는 것이므로 가장 정확한 최적의 경로를 찾을 수 있지만 많은 계산량에 따른 시간이 많이 소요된다. 반면에 유전알고리즘은 사용하는 유전 연산자 자체가 국소점을 벗어날 수 있는 방안을 제공하고 복수 계산 방식보다는 적은 계산량을 가진다.

기존에 연구되어온 유전 알고리즘을 살펴보면 유전연산이 미리 작업을 할 환경에 대한 지도와 같은 완전한 정보를 가지고 있는 상태에서 이루어지는 경우가 대부분이었으며 이러한 환경 정보를 바탕으로 x,y의 좌표 쌍으로 이루어진 염색체로서 구성된 유전자형을 사용하여 유전 알고리즘을 적용하였다[5][6]. 이러한 유전지형은 그 길이가 일정하지 않아 유전 연산시에 일정한 길이를 갖는 유전연산시 유전자형이 일정한 경우보다 연산과정이 복잡해지게 된다.

본 논문에서는 이러한 문제점을 보완하기 위하여 몇 가지 가정을 통하여 새로운 유전자형을 제안하고 이 새로운 유전자형을 갖는 유전 알고리즘을 자율이동 로봇의 경로 계획에 적용하여 본 알고리즘이 최적 경로 생성이 가능하고 적응성도 뛰어난 시뮬레이션을 통하여 보인다.

2. 본 론

$$\Delta y = 0 \quad (7)$$

$$\Delta \theta = 0 \quad (8)$$

2.1 이동로봇의 좌표값 설정

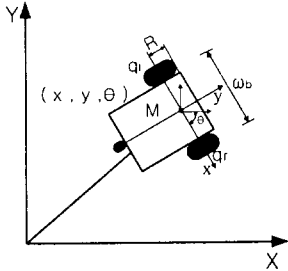


그림 1 이동로봇의 좌표계
Fig1. Coordinate of the mobile robot

로봇은 두 개의 구동용 바퀴와 하나의 무동력 회전 바퀴를 사용하는 구조를 가지고 있다. 그림(1)는 이동로봇의 절대 좌표(Global coordinate)와 로봇자체 좌표(Robot coordinate)를 나타내고 있다.

이동로봇은 2차원 x-y 평면위를 움직이므로 (x,y)의 좌표를 사용하여 현재의 위치를 나타내고 다음 위치로 이동할 방향을 알기 위해서는 현재 로봇이 놓여 있는 방향 (θ)을 알아야 한다. 비홀로노믹 제한 조건이 있는 일반적인 바퀴 구동형 이동로봇의 운동방정식은 다음과 같이 나타내어진다.

$$\dot{x} = v \cos \theta \quad (1)$$

$$\dot{y} = v \sin \theta \quad (2)$$

$$\dot{\theta} = \omega \quad (3)$$

$$v = R(\dot{q}_r + \dot{q}_l)/2 \quad (4)$$

$$\omega = R(\dot{q}_r - \dot{q}_l)/\omega_b \quad (5)$$

여기서 q_r 과 q_l 은 각 오른쪽과 왼쪽 바퀴의 직선 속도(바퀴의 회전속도×바퀴의 반경)이고 ω_b 는 두 바퀴 중심선 사이의 거리를 나타낸다.

이동로봇의 이동궤적은 직선궤적과 원호 궤적으로 나누어 질 수 있으며 다음과 같은 식으로 나타낼 수 있다.

• 직선 궤적 운동

$$\Delta x = v \Delta t \quad (6)$$

• 원호 궤적운동

$$\Delta x = \rho \sin \Delta \theta \quad (9)$$

$$\Delta y = \rho(1 - \cos \Delta \theta) \quad (10)$$

$$\Delta \theta = v \Delta t (1 - \cos \Delta t) / \rho \quad (11)$$

$$\Delta X = \Delta x \cos \theta - \Delta y \sin \theta \quad (12)$$

$$\Delta Y = \Delta x \sin \theta + \Delta y \cos \theta \quad (13)$$

$$X = \sum \Delta X \quad (14)$$

$$Y = \sum \Delta Y \quad (15)$$

여기서 ρ 는 가상의 원점에서의 회전 반경이며 $\Delta \theta$ 는 이동로봇의 회전 변위각, Δt 은 이동로봇이 일정 지점까지 이동하는데 걸린 시간을 나타낸다.

2.2 유전 알고리즘

유전 알고리즘은 자연계의 선택과 유전의 현상을 기반으로 한 검색 알고리즘이다[3]. 주어진 문제를 풀기 위해 일정한 형태의 초기 개체군을 도태로 일정 규칙에 의해 각 개체들을 변경하고 자연계에서의 '다윈의 법칙'과 같이 우수 형질의 개체는 선택(selection)하고 열성형질의 개체는 도태(reproduction)시켜 최적의 해를 형성시키는 것이다[3].

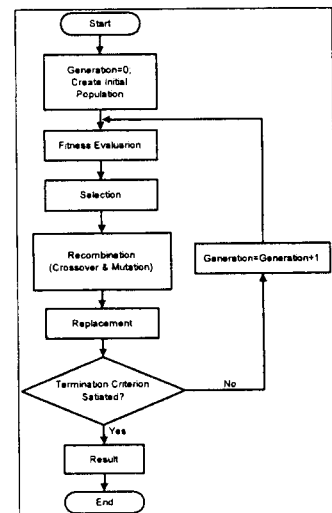


그림 2 유전 알고리즘의 흐름선도
Fig2 Flowchart of Genetic Algorithms

는 유전자가 나타내는 좌표 α 에서 $\alpha-1$ 좌표까지는 현재의 위치에서 수직으로 주행을 하고 $\alpha-1$ 에서 α 로 장애물의 모서리 부분이 아닐 경우 대각선 주행을 하고 모서리일 경우 이동로봇의 폭을 감안하여 직각 주행을 하게 된다.

제안한 유전자의 길이가 항상 일정해지므로 전체 유전자의 길이와 개체군이 차지하는 공간은 다음과 같이 계산할 수 있다. 전체 개체군이 차지하는 공간과 개체의 유전자 표현형의 길이는 각 유전자가 차지하는 2진 수열의 비트 수에 의해서 결정되어진다. 하나의 유전자가 차지하는 비트의 수 (l_b)는

$$l_b = 2 + \lceil \log_2 m \rceil \text{ bits} \quad (16)$$

이고, 여기서 m 은 세로격자의 개수이다. 그리고 전체 표현형의 길이(l_s)와 전체 개체군의 공간 (m_p)는

$$l_s = (n-1) \times (2 + \lceil \log_2 m \rceil) \text{ bits} \quad (17)$$

$$m_p = p \times ((n-1) \times (2 + \lceil \log_2 m \rceil)) \text{ bits}$$

이다. 여기서 n 은 가로격자의 개수이고 p 는 전체 개체군의 개체의 수이다. 그리고 $\lceil x \rceil$ 는 x 를 넘지 않는 최대의 정수를 나타낸다.

한 격자가 나타내는 실제 거리를 d 라고 하면, 대각선 주행과 직선후 대각선 주행으로 나뉘어지는 현재 위치하고 있는 격자(g_{now})에서 인접한 격자(g_{next})로의 이동시의 실제 거리는 다음과 같이 나타내어진다.

- 첫 번째 비트 0 :

$$\sqrt{1 + (g_{now} - g_{next})^2} \times d \text{ (meters)} \quad (18)$$

- 첫 번째 비트 1 :

$$((g_{now} - g_{next} - 1) + \sqrt{2}) \times d \text{ (meters)} \quad (19)$$

한 개체가 나타내는 경로의 길이는 위의 (18),(19)식에 의하여 계산된 각 부분 경로의 거리의 합(f)으로 나타내어진다.

2.2.3 적합도 함수

적합도 함수는 유전 알고리즘의 성능에 큰 영향을 미치는 한 요소 작용을 한다. 먼저 목적함수와 적합도 함수는 식(20)과 같이 정의한다.

$$y = (n-2+\sqrt{2}) * m - f + \sqrt{m^2 + n^2}$$

$$fit = a / add$$

$$\left(\begin{array}{l} a_i = y_i - \min(y) \\ add = \sum_{i=0}^{m-1} a_i \end{array} \right) \quad (20)$$

여기서, y 는 각 개체가 갖는 목적함수 값이며 n 과 m 은 각각 가로격자와 세로격자의 수를 나타내고, f 는 실제 이동거리를 나타낸다. 개체의 적합도는 시작점과 목표점 사이에 장애물이 없는 경우의 최단 경로인 $\sqrt{m^2 + n^2}$ 을 빼줌으로서 최단경로의 목적함수의 값이 가장 큰 값을 갖도록 하였다.

fit 는 적합도를 나타내고 전체 개체군의 개체들을 최소 목적함수의 값으로 뺀 값(a_i)을 그 값들의 합(add)으로 나눈 값으로 한다.

이러한 방법으로 적합도 함수를 취함으로서 0과 1사이의 값으로 모든 개체의 적합도를 나타내고 이를 바탕으로 룰렛-휠(roulette wheel) 선택법을 적용하여 유전 연산을 할 개체를 선택한다.

2.2.3 유전 연산자

본 논문에서는 경로계획을 위한 유전 연산자로 두 가지를 사용한다. 첫 번째 연산자로는 한 점 교배(1-point-crossover)를 사용한다. 이때 미리 정하여둔 교배가 일어날 확률 즉, 교배율(crossover-rate)에 의하여 교배를 제한한다. 교배를 연산을 수행할 때 기존의 유전자형을 사용할 경우 각 유전자의 길이가 다르기 때문에 교배점을 정하는 것이 까다로운 반면 본 논문에서 제안한 유전자형을 사용할 경우 이러한 문제를 고려하지 않고 연산을 수행할 수 있게 된다.

두 번째의 연산자로는 돌연변이(mutation)로 한 개체를 택하여 임의의 한 비트를 바꾸어 버림으로서 전혀 다른 형질을 가질 수 있도록 한다. 돌연변이는 일어날 확률을 돌연변이율(mutation-rate)에 의해 제한하여 필요 이상의 연산이 수행되는 것을 막는다. 하지만 이러한 돌연변이 연산을 적절하게

허용함으로써 통하여 국소점에 빠진 경우에도 빠져나올수 있는 방법을 제공하고 빠르게 최적의 경로를 찾을 수 있게 한다.

2.2.3 경로 추적 및 재설정

유전자 알고리즘에 의하여 생성된 경로는 장애물을 고려하지 않고 찾은 경로이므로, 처음에는 개체군을 생성한 다음 장애물이 없을 때의 최단 경로인 대각선 직선 경로를 이동로봇이 추적할 때 앞에서 가정한 것과 같이 사방의 장애물의 유무를 판단하고 장애물을 발견했을 경우 주어진 경로를 따라가지 않고 선회하여 장애물을 회피한다. 그리고 장애물이 없는 위치에 도착하면 그 위치에서 다음의 원래의 경로로 복귀하여 계속 진행하도록 한다. 이러한 과정에서 각 다음 위치로 이동하는 도중 장애물을 발견하면 선회한 경로로써 원래의 경로를 재설정할 수 있도록 하여 미리 알고 있지 않은 환경에서도 제안한 알고리즘의 적용이 가능하게 하고 시간적으로 변화하는 환경에서도 보다 강인한 특성을 가질 수 있으며 빠르게 최적의 경로를 찾을 수 있다.

위에서와 같은 재설정되는 유전자가 다시 생성되는 것을 방지하기 위하여 일단 염색체가 장애물이 있는 곳을 나타낸다는 것을 알게 되면 전체 개체군에서 그와 같은 염색체를 갖는 모든 다른 유전자들을 수정하여 다음 세대에 나타나는 것을 최소화한다. 이후에는 오프라인(off-line)으로 유전 연산을 여러번 수행한 후 생성된 최적의 경로를 다시 이동로봇이 경로 추적을 함으로써 다시 장애물이 있는 지점의 염색체를 다른 값으로 교체해 나가는 과정을 여러번 반복하여 최적의 경로를 재설정한다.

3. 시뮬레이션

본 논문에서 16×16 격자를 갖는 작업환경에서 좌측 상단을 시작점으로 하고 우측 하단 격자를 목표점으로 하여 여러 장애물의 형태를 갖는 경우를 시뮬레이션을 하였다.

개체군의 크기는 20으로 하였고, 연산자는 앞에서 언급한 것과 같이 1-point-crossover, 돌연변이를 사용하고 각 연산이 일어날 확률은 교배가

0.8, 돌연변이는 0.04 ~0.05로 하고 부모 개체의 선택 방법은 roulette wheel 선택법을 사용하였다. 위의 수치들은 시뮬레이션을 통한 시행착오법에 의해서 구하였다. 다음의 각 그림들은 환경이 변화함에 따른 경로의 변화를 보여주고 시작점은 'a' 으로 목표점은 'b'로 표시하고, 장애물이 있는 영역은 검정색으로 그 영역의 격자를 표현하였다. 장애물의 모서리 부분에서는 이동로봇의 회전시 충돌 가능성이 있으므로 대각선이 아닌 직각으로 주행하게 하였다.

그림 5, 6는 작업환경에 고정된 장애물이 위치해 있는 경우 유전 연산 실행 전의 대각선 직선 경로를 이동한 이동로봇이 이동한 후 변경된 경로와 20번의 유전연산에 의하여 형성된 최단경로를 이동로봇이 이동한 후 변경된 경로 나타낸다.

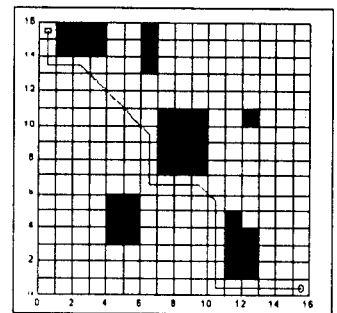


그림 5 초기의 대각선 주행 후 생성된 경로
Fig 5 Generated path after moving initial diagonal path

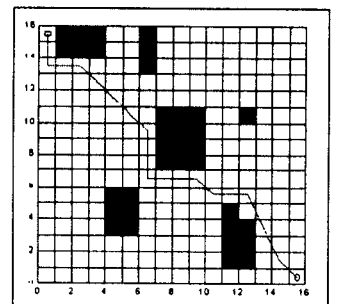


그림 6 유전연산 20번 이후의 생성된 경로
Fig 6 Generated path after operating 20 times genetic operation

그림 7, 8, 9은 보다 복잡한 작업환경의 경우와 이런 환경 하에서 시간적으로 가변인 장애물이 있는 경우와 없어진 경우의 최적 경로의 변화를 보이

고 있다.

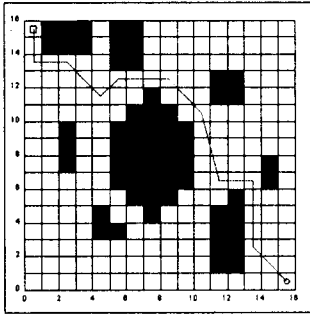


그림 7 환경이 복잡한 경우의 경로생성
Fig 7 Path generation of complex environment

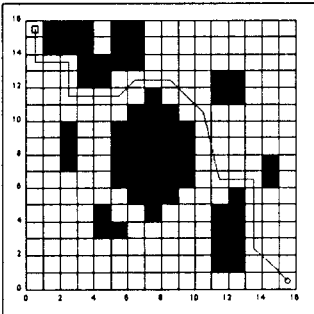


그림 8 장애물이 추가된 경우
Fig 8 Added an obstacle

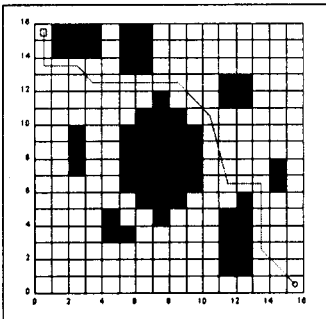


그림 9 장애물이 제거된 경우
Fig 9 Removed the obstacle

5. 결론 및 고찰

본 논문에서 제안한 유전 알고리즘을 이용한 경로 계획은 작업환경에 대한 사전의 정보 없이도 최적의 경로 계획을 할 수 있음을 보이고, 시간적

으로 가변인 장애물이 있는 경우에도 강인한 특성을 지님을 보였다.

제안된 알고리즘은 복잡한 환경에서 장애물이 많이 존재하는 환경 하에서도 강인한 특성을 가지는 것을 시뮬레이션을 통하여 알 수 있었다. 그리고 단순한 장애물이 많지 않은 환경 하에서는 장애물이 없을 경우의 경로를 최초 이동함으로서 빠르게 최적의 경로를 찾아간다. 또한, 시간적으로 변화하는 장애물을 만났을 경우에도 최적의 경로 계획을 잘하는 것을 볼 수 있으며 이 장애물이 없어졌을 때에도 원래의 최적 경로에 가깝게 복귀함을 알 수 있었다.

본 논문에서의 시뮬레이션 결과를 바탕으로 향후 실제의 이동로봇에 적용하여서도 위와 같은 성능을 가질 수 있는지를 고찰해 보아야 할 것이다.

참고문헌

- [1] K. Sugihara, "GA-based on-line path planning for SAUVIM," *Lecture Notes in Artificial Intelligence*, vol.1416, pp.329-338, Springer-Verlag, 1998.
- [2] K. Sugihara, "Measures for performance evaluation of genetic algorithms," Proc. 3rd Joint Conference on Information Sciences, Research Triangle Park, NC, vol.I, pp172-175, March 1997.
- [3] D.E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning," Addison Wesley, 1989
- [4] 이준배, "유전자 알고리즘을 이용한 전체/다목적 최적화 알고리즘의 개발," 한국과학기술원 박사학위 논문, 1996.
- [5] 광한택, 이기성, "유전 알고리즘을 이용한 이동로봇의 경로 계획," 한국 자동제어공학회, Proc. 11th KACC, 1996
- [6] 김진수, 이영진, 장용훈, 이권순, "유전알고리즘을 이용한 이동로봇의 경로계획 및 충돌회피에 관한 연구," 한국 자동제어공학회, Proc. 11th KACC, 1996