

Firefly Algorithm 을 이용한 군집 이동 로봇의 경로 계획

Path Planning of Swarm Mobile Robots Using Firefly Algorithm

김 휴 찬, 김 제 석, 지 용 관, 박 장 현*
(Hue-Chan Kim¹, Je-Seok Kim², Yong-Kwan Ji³, and Jahng-Hyon Park²)

¹Machine Design Team, Canon Korea Business Solution

²Department of Automotive Engineering, Hanyang University

³ADAS Engineering Team, Hyundai Mobis

Abstract: A swarm robot system consists of with multiple mobile robots, each of which is called an agent. Each agent interacts with others and cooperates for a given task and a given environment. For the swarm robotic system, the loss of the entire work capability by malfunction or damage to a single robot is relatively small and replacement and repair of the robot is less costly. So, it is suitable to perform more complex tasks. The essential component for a swarm robotic system is an inter-robot collaboration strategy for teamwork. Recently, the swarm intelligence theory is applied to robotic system domain as a new framework of collective robotic system design. In this paper, FA (Firefly Algorithm) which is based on firefly's reaction to the lights of other fireflies and their social behavior is employed to optimize the group behavior of multiple robots. The main application of the firefly algorithm is performed on path planning of swarm mobile robots and its effectiveness is verified by simulations under various conditions.

Keywords: firefly algorithm, mobile robot, convex hull, swarm robots

I. 서론

다수의 로봇이 각자가 가진 능력만으로 해결하기 어려운 작업을 주위의 로봇들 간에 상호작용과 협력을 통해 주어진 작업을 원활하게 수행할 수 있도록 구성된 시스템을 군집로봇 시스템이라고 한다[1,2]. 군집 로봇에서 각각의 단일 로봇을 개체(agent)라 하는데 개체라는 것은 주어진 일과 환경에 대해 상호 작용을 하면서 환경을 인지하고 상황에 맞는 상황 판단과 의사 결정을 한 후 행동을 하여 환경에 반응을 하는 모든 것들로 정의된다[3]. 이러한 군집 로봇 시스템을 이용하면 한 대의 로봇을 이용하여 동일한 작업을 수행하는 시스템과 비교할 때, 로봇 1대의 고장 또는 파손에 의해 전체 작업 계획이 받게 되는 손실이 비교적 적으며 로봇의 교체 및 수리가 용이하고 보다 복잡한 작업 수행이 가능하다.

군집 로봇 시스템에서 주어진 목표를 달성하기 위한 필수 요소는 로봇들 간의 협력적인 공동 작업을 위한 전략이다. 최근에 들어, 군집 지능이론(swarm intelligence) [4-6]은 집단 로봇 시스템의 설계화 구현에 새로운 접근 방법을 제공함으로써 상대적으로 간단한 구조를 가진 다수의 로봇들의 상호 협력을 통해 견실하고 유연한 방법으로 복잡한 임무를 수행할 수 있는 군집 로봇 시스템(swarm robotic system)의 구성을

가능하게 하였다. 군집 이론 중 하나인 FA (Firefly Algorithm) [7,8]은 반딧불이의 불빛을 내는 행동과 사회적 행동 양식에 기반을 둔 개체군 기반의 최적화 방법으로 주어진 조건하에서 목표를 향해 이동하는 기본적인 목적과 동일하며, 군집 지능의 적용을 용이하게 하는 장점이 있다.

본 논문에서는 다수의 로봇이 목표점에 도달하고, 주어진 작업을 성공적으로 수행하기 위한 시스템을 구현하기 위하여 군집 지능의 모델인 FA를 이용하여 군집 이동 로봇의 경로 계획에 대해서 연구를 수행하였다. 본 논문의 구성은 다음과 같다. II 장에서는 FA를 이용한 군집 이동 로봇의 경로 계획에 대한 주요 탐색 성능의 결정 요소들에 대해서 설명하고, III 장에서는 군집 이동 로봇의 경로에 대한 최단 경유점 생성 방법에 대해서 설명한다. IV 장에서는 생성된 경유점을 이용한 군집 이동 로봇의 경로 계획에 대해서 설명하고 V 장에서 이를 시뮬레이션을 통하여 제안한 알고리즘을 검증한다.

II. 반딧불이 알고리즘(FA: Firefly Algorithm)

1. FA의 개요

지구 상에 있는 대부분의 반딧불이는 짧고 규칙적인 불빛을 만든다. 이 깜박거리는 불빛은 짝이 되는 반딧불이를 유혹하거나 먹이를 유인할 때 사용된다. 규칙적인 불빛에서 깜박임률과 깜박거리는 시간은 반딧불이의 수컷과 암컷 모두가 가지는 신호적인 시스템으로서 불빛 강도 I 는 특정 거리 r 이 증가함에 따라 $I \propto 1/r^2$ 에 비례하여 감소한다. 그리고 공기는 거리가 증가함에 따라 불빛을 흡수하여 흐리게 된다. 이러한 두 가지가 병합된 요소는 반딧불이로 하여금 제한된 거리에서 서로를 볼 수 있게 만든다. 깜박거리는 불빛을 목적함수(objective function) 형태로 연관시켜 공식화하고 최적화한 것이 FA (Firefly Algorithm)이다[9].

FA는 다음과 같은 3 가지의 이상적인 규칙을 따른다. ①

* 책임저자(Corresponding Author)

Manuscript received February 20, 2013 / revised March 10, 2013 / accepted March 15, 2013

김휴찬: 캐논코리아 비즈니스 솔루션 기구설계팀

(hckim3@canon-bs.co.kr)

김제석: 한양대학교 자동차공학과(n4clover@hanyang.ac.kr)

지용관: 현대모비스 ADAS설계팀(ykji@mobis.co.kr)

박장현: 한양대학교 미래자동차공학과(jpark@hanyang.ac.kr)

* 본 논문은 지식경제부 산업원천기술개발사업(10033555, 고밀도 작업용 멀티로봇 공정 효율화 및 공통기술 개발)과 지식경제부 우수 제조기술연구센터 기술개발사업(10031484, 차세대 협업 생산 로봇을 위한 다자유도 Robot ARM 및 응용기술 개발)의 일환으로 수행되었음.

모든 반딧불이는 암수 구별이 없다. ② 적합도(tractiveness)는 밝기에 비례한다. 만약 2 마리 반딧불이가 있는 경우 상대적으로 덜 밝은 반딧불이가 더 밝은 반딧불이로 이동한다. 그러나 주변에 주어진 밝기보다 밝은 반딧불이가 없다면 임의로 움직인다. ③ 반딧불이의 밝기는 목적함수에 의해서 결정되거나 영향을 받는다. 밝기는 유전 알고리즘의 적합도 함수와 유사하게 정의된다[6].

2. 밝기 강도와 적합도(tractiveness)

FA는 2가지의 중요한 요소가 있다. 하나는 밝기 강도이며, 또 다른 하나는 적합도의 공식화이다. 반딧불이의 적합도는 목적함수와 연관된 밝기에 의해서 결정된다. 최적화 문제의 경우 특정 위치 x 에서 반딧불이의 밝기 강도 I 는 $I(x) \propto f(x)$ 로 선택된다. 그러나 적합도 β 는 다른 반딧불이의 눈에 의해서 주시나 판단이 된다.

반딧불이의 i 와 j 의 거리는 r_{ij} 로 정의하고, 밝기 강도는 거리에 따라 감소되므로 불빛 강도 $I(r)$ 은 역제곱법칙에 따라 식 (1)과 같이 정의한다.

$$I(r) = \frac{I_s}{r^2} \quad (1)$$

여기서 I_s 는 특정 장소에서의 불빛 강도이다. 또한 밝기 강도는 대기 중에서 흡수되므로 이를 수치적으로 γ 로 표현하고, 이는 일정거리를 벗어난 개체끼리는 볼 수 없다. 적합도에 고정된 빛의 흡수 계수 γ 를 가진 어두운 곳에서 불빛 강도 I 는 식 (2)와 같이 정의한다.

$$I(r) = I_0 e^{-\gamma r^2} \quad (2)$$

여기에서 I_0 는 초기 불빛 강도이다. I_s/r^2 에서 $r=0$ 일 때 특이점을 피하기 위해서, 역제곱법칙과 흡수율의 조합된 결과는 식 (3)과 같은 가우시안 방식으로 표현된다.

$$I(r) = \frac{I_0}{1 + \gamma r^2} \quad (3)$$

매우 짧은 거리인 경우, $r=0$ 에 관해 식 (4)와 같이 전개가 가능하기 때문에 식 (2)와 식 (3)은 근사적으로 식 (5)의 등식이 성립한다[6].

$$e^{-\gamma r^2} \approx 1 - \gamma r^2 + \frac{1}{2} \gamma^2 r^4 + \dots, \quad (4)$$

$$\frac{1}{1 + \gamma r^2} \approx 1 - \gamma r^2 + \gamma^2 r^4 + \dots$$

$$I(r) = I_0 e^{-\gamma r^2} = \frac{I_0}{1 + \gamma r^2} \quad (5)$$

반딧불이의 적합도는 근접한 반딧불이에 의해 보여지는 불빛 강도에 비례하기 때문에 적합도 β 를 식 (6)과 같이 정의할 수 있다.

$$\beta(r) = \beta_0 e^{-\gamma r^2} \quad (6)$$

β_0 는 $r=0$ 일 때 적합도 인자이다. 이것은 전형적인 함수보다 $1/(1+r^2)$ 의 계산량이 적기 때문에 식 (6)은 식 (7)과

같이 표현할 수 있다.

$$\beta(r) = \frac{\beta_0}{1 + \gamma r^2} \quad (7)$$

식 (6)와 식 (7)은 적합도의 변화가 β_0 에서 $\beta_0 e^{-1}$ 가 되는 거리를 특정거리 $\Gamma = 1/\sqrt{\gamma}$ 과 같이 정의한다.

적합도 함수 $\beta(r)$ 은 식 (8)과 같이 단조롭게 감소하는 함수이다.

$$\beta(r) = \beta_0 e^{-\gamma r^2}, \quad (m \geq 1) \quad (8)$$

고정된 γ 에 대해 특정 거리 Γ 는 식 (9)와 같다.

$$\Gamma = \gamma^{-1/m} \rightarrow 1, \quad m \rightarrow \infty \quad (9)$$

최적화 문제 안에서 주어진 거리 변수 Γ 에 대해 흡수 계수 γ 는 전형적인 초기값으로 사용된다. 즉,

$$\gamma = \frac{1}{\Gamma^m} \quad (10)$$

특정 장소 x_i 와 x_j 에서 두 반딧불이 i 와 j 사이의 거리는 식 (11)과 같이 표현된다.

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^d (x_{i,k} - x_{j,k})^2} \quad (11)$$

여기서 $x_{i,k}$ 는 i 번째 반딧불이의 좌표에 대한 k 번째 성분이다.

i 번째 반딧불이의 움직임은 더 밝은 j 번째 반딧불이에 의해서 식 (12)와 같이 결정된다.

$$x_i = x_i + \beta_0 e^{-\gamma r^2} (x_j - x_i) + \alpha (\text{rand} - 0.5) \quad (12)$$

여기서 $\beta_0 e^{-\gamma r^2} (x_j - x_i)$ 는 적합도 요소이고, rand 는 $[0, 1]$ 중에서 추출된 임의의 수이다. 그리고 $\beta_0 = 1$ 이고, $\alpha \in [0, 1]$ 이다.

고정된 빛의 흡수계수 γ 는 수렴의 속도를 결정하고, FA의 수렴하는 행동을 결정한다. 이론적으로는 $\gamma \in [0, \infty]$ 이지만, 실제 환경에서 $\gamma = O(1)$ 은 최적화된 시스템의 특정거리 Γ 에 의해서 결정된다. Γ 의 범위는 0.1에서 10까지 이다.

III. 최적 경유점 생성

1. Convex Hull Path의 생성

본 논문에서 최단 경유점을 찾기 위해 Convex Hull [10]을 이용하여 직선 경로를 기준으로 직선 경로에 교차하는 장애물을 검색하여 이를 회피하는 경로를 생성한다. 우선 그림 1(a)와 같이 직선경로를 기점으로 Upper와 Down의 2개의 영역으로 그림 1(b)와 같이 분리한다[11].

Upper와 Down으로 분리된 두 영역에서 시작지점과 목표지점을 기준으로 각각 Convex Hull을 취하게 되면 식 (13)과 같이 시작지점에서 목표지점까지 갈 수 있는 4가지의 경로 중에서 가장 짧은 2개의 Convex Hull Path를 얻을 수 있다. 이를 후보 Convex Hull Path라 한다.

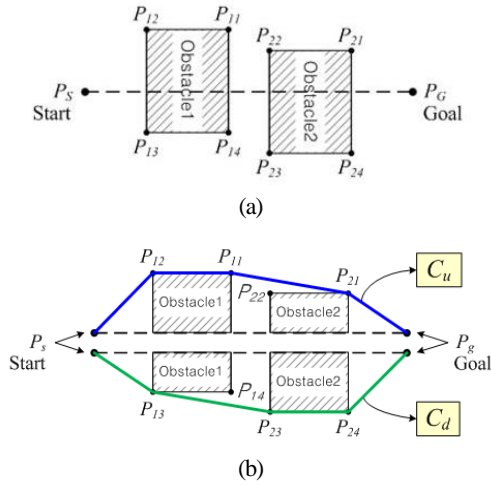


그림 1. 최적 경유점(convex hull path) 생성.
Fig. 1. Generation of convex hull path.

$$C_u = [P_s \ P_{11} \ P_{12} \ P_{21} \ P_{22} \ P_g]^T$$

$$C_d = [P_s \ P_{13} \ P_{14} \ P_{23} \ P_{24} \ P_g]^T$$
(13)

2. 내부 장애물과 외부 장애물

앞 절에서 생성한 Convex Hull Path는 장애물 외부를 지나 는 경로만이 고려되었다. 최적의 경로가 항상 장애물의 외부 로 설정되는 것이 아니기 때문에 내부로 생성되는 경로도 고 려해야 한다. 내부 장애물은 직선 경로에 교차하는 장애물의 꼭지점 중에서 어느 하나라도 Convex Hull Path를 구성하는데 사용되지 않는 장애물이다.

그림 2와 같은 환경에서는 장애물과 장애물 사이를 가로 지르는 경로를 생각할 수 있다. 그래서 Convex Hull Path를 생 성한 다음 C_u와 C_d에 대해서 내부장애물의 존재여부를 검 사한 뒤 회피하는 경로를 생성한다. 우선 그림 2(a)와 같이 C_d의 내부장애물에 대해서 직선경로를 기준으로 Convex Hull 을 구성하고 Line Segment 중에서 장애물과 교차하는 L₃를 기준으로 그림 2(b)와 같이 Convex Hull을 재구성한다. 두 개 의 Convex Hull 중에 최적 경로를 선택하고, 그림 2(c)와 같이 기존의 경로와 결합하면 내부의 장애물을 회피하는 경로를 생성할 수 있다. C_u와 C_d에서 검색되는 내부장애물에 대한 회피 경로 생성이 완료되면 그림 2(d)와 같이 총 세 가지의 Convex Hull Point가 생성된다.

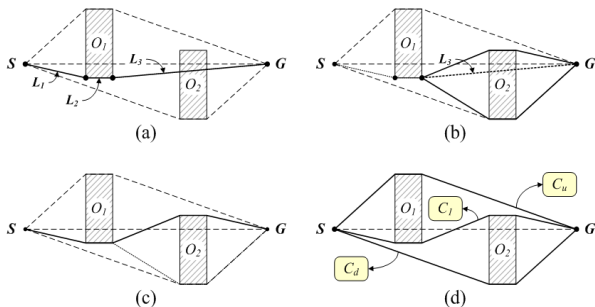


그림 2. 내부 장애물의 경로 계획.
Fig. 2. Path planning about inner obstacles.

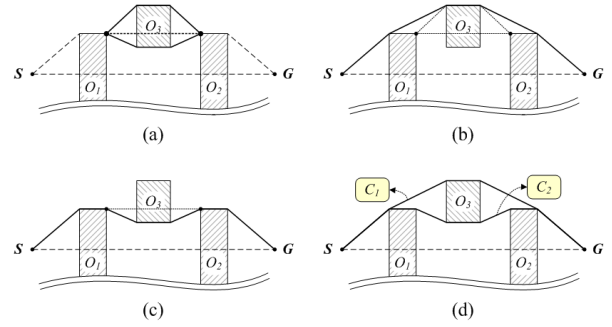


그림 3. 외부 장애물의 경로 계획.
Fig. 3. Path planning about outer obstacles.

내부장애물의 존재여부를 확인한 후 생성한 여러 가지의 Convex Hull Path에 또 다른 장애물인 외부장애물이 존재할 수 있다. 외부장애물은 직선 경로에 교차하지 않으면서, Convex Hull Path에 교차하는 장애물을 Convex Hull Path의 외 부장애물로 정의한다.

외부장애물의 회피 과정은 그림 3과 같다. 그림 3(a)와 같 이 L₃를 직선 경로로 간주하고 상단부분과 하단부분으로 Convex Hull을 생성한다. 상단부분은 다시 본래의 직선 경 로(시작지점과 목표지점)를 기준으로 Convex Hull을 재 생 성하여 장애물에 대해 가장 바깥 방향으로 경로를 생성한 다. 하단부분은 그림 3(c)와 같이 기존 경로에서 L₃를 삭제 하고, 생성한 Convex Hull을 더해서 장애물을 회피하도록 경로를 생성한다. 위 과정을 거쳐 최종적으로 그림 3(d)와 같이 외부장애물을 회피하는 2가지의 Convex Hull Path를 획득할 수 있다.

3. Convex Hull Path의 결정 방법

지금까지 Convex Hull Path의 생성에 대해 설명하였고, 임의 의 환경에서 여러 개의 Convex Hull Path가 생성되는 것을 보 았다. 생성된 여러 Convex Hull Path중에서 가장 좋은 경유점 을 선택하기 위해서 경로의 길이(distance of path)와 방위변화 량(change of orientation angle)을 사용하여 비교한다[11].

n 개의 Convex Hull Path C_i(i=1,2,...,n)가 있을 때, 경로 의 길이 D_i는 식 (14)와 같이 절대거리를 나타내고, 방위변화 량 A_i는 식 (15)와 같이 표현한다.

$$D_i = \sum_{k=1}^m |C_i(k) - C_i(k-1)|$$
(14)

$$A_i = \sum_{k=1}^m |\theta_i(k) - \theta_i(k-1)|$$
(15)

방위변화량은 그림 4와 같이 나타내고, 값이 작을수록 직 선에 가까운 경로를 뜻하기 때문에 경로의 길이와 함께 경로 를 판별할 때 중요한 인자로 사용된다[11].

경로의 판별은 식 (16)과 같이 나타내며, 이 중에서 가장 작은 F_i를 갖는 경로 C_i를 최종 Convex Hull Path로 결정한다.

$$F_i = \frac{D_i}{\max(D_i)} + \omega \frac{A_i}{\max(A_i)}$$
(16)

Convex Hull Path 판별식에서 방위변화량에 가중치 ω를 부 여하여 전체 판별식에서 차지하는 비중을 선택할 수 있다.

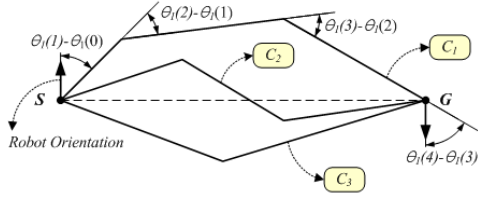


그림 4. 방위변화량.
Fig. 4. A change of orientation.

IV. 군집 이동 로봇의 경로 계획

1. 군집 로봇 시스템과 FA의 비교

군집 로봇을 이용한 목표점 탐색 작업은 최적화 문제에서 탐색 공간상의 전역 최적해를 찾는 과정과 동일하다. 이러한 관점에서 군집 로봇의 목표점 탐색과 FA는 비교적 수월하게 결합이 가능하며, FA를 적용한 로봇 탐색 작업은 개체군 기반의 최적화 방법의 특징을 가지게 된다.

본 논문에서는 기존의 연구 사례들과 동일하게 FA에서 로봇 군집을 구성하는 로봇은 '개체군'을 구성하는 개체가 된다. FA의 진행 절차에 따라 초기로봇들은 탐색 공간과 이동거리의 구속 조건하에서 탐색공간에 위치하게 되며, 적합도 평가에 따라 매 반복 세대에 갱신되는 개체들은 로봇들이 이동해야 할 다음 탐색 지점으로 결정된다. 군집을 구성하는 로봇들의 탐색 과정은 종료 조건을 만족할 때까지 위의 과정을 반복하게 된다.

2. FA를 이용한 군집 이동 로봇의 경로계획

본 논문에서 제안한 FA를 이용한 군집 이동 로봇의 경로 계획은 주어진 공간에서 식 (17)과 같이 목적함수를 최소화 하는 방식으로 이동한다[6].

$$\text{if } x \in S \subset \mathbb{R}^n, f(x^*) = \min_{x \in S} f(x) \quad (17)$$

FA를 이용한 군집 로봇의 경로계획의 구성은 다음과 같다.

Step 1: n 차원 해 공간상에서 초기 군집 m 개의 개체의 시작점과 목표점을 정의한다.

Step 2: 경유점(convex hull path)과 장애물에 목적함수를 발생시킨다. 식 (18)에서 $f_p(x)$ 는 경유점의 목적함수를 나타내며, $f_o(x)$ 는 장애물의 목적함수를 나타낸다. 그리고 x_i 는 탐색공간상의 좌표를 나타낸다.

$$f(x) = f_o(x_i) - f_p(x_i), x_i = [x_{i,1}, \dots, x_{i,n}]^T \quad (18)$$

Step 3: 군집적합도 판정기준은 로봇들이 다수의 목적지로 이동해야 할 경우 같은 경유점을 지나가는 군집의 결합 및 분할을 결정하는 기준이다. 군집 적합도 판정 기준에 따라 군집을 구성한다. 군집 적합도 판정 기준은 다음과 같다.

- 1) 동일한 경유점 또는 목표점을 가지고 있다.
- 2) 개체간의 거리가 일정범위 안에 있다.

위의 조건을 만족하지 않으면 군집은 따로 구성된다. 여기서 조건 2는 식 (19)와 같이 표현된다. 여기서 R 은 일정범위를 나타낸다.

$$\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \leq R \quad (19)$$

Step 4: 각 개체들에 대해 적합도를 평가한다. 군집의 각

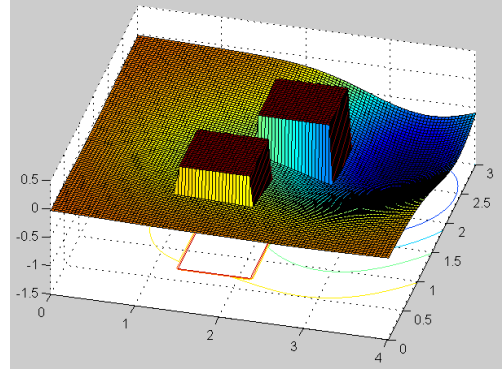


그림 5. 목적함수.
Fig. 5. Objective function.

개체는 목적함수에 의해 평가되고, 이들 모두는 $f(x_i)$ 로 설정한다. 각 개체들의 적합도를 그 개체의 $f(x_i)$ 와 비교하여 적합도가 더 크면 그 값이 $f(x_i)$ 가 되고, 리더 개체가 된다.

Step 5: 식 (20)에 따라 개체군 x_j ($j=1, \dots, m$)에 대해 $f(x_j) > f(x_i)$, $j \neq i$ 인 경우 리더 개체를 제외한 개체의 위치를 계산하여 업데이트한다. 새로운 위치정보를 가진 개체들은 목적함수에 의해 평가된다. 각 개체들의 적합도는 이전 $f(x_i)$ 와 비교되며, $f(x_i)$ 를 재설정한다.

$$x_i(k+1) = x_i(k) + \beta_0 \cdot e^{-\gamma \tau_0^2} \cdot (x_j(k) - x_i(k)) + u_i + U_{Tot} \quad (20)$$

식 (20)에서 $x_i(k)$ 는 탐색 공간을 움직이는 개체로 현재 위치 정보에 대한 적합도를 구하고 최적의 위치정보와 비교하기 위한 값이다. $X_j(k)$ 는 모든 개체 중에서 가장 최적의 위치정보를 나타내는 개체로 주어진 문제의 최적해가 된다.

$$\gamma = \frac{\gamma_0}{r_{\max}}, \gamma_0 \in [0, 1] \quad (21)$$

$$r_{\max} = \max d(x_i, x_j), \forall x_i, x_j \in S$$

또한 $u_{i,k}$ 는 탐색 공간 차원 k 에서 개체의 위치를 임의로 이동하게 하는 요소로 식 (22)와 같이 생성한다[13].

$$u_{i,k} = \alpha \cdot \left(\text{rand} - \frac{1}{2} \right) \quad (22)$$

$$u_{i,k} \leftarrow (\min u_{i,k}, \max u_{i,k})$$

α 는 임의의 값에 대한 가중치 인자이다. rand_2 는 $[0, 1]$ 사이에서 균등하게 분포하는 임의의 벡터이다. 마지막으로 U_{Tot} 는 포텐셜 필드의 값으로 Step 6에서 설명한다.

Step 6: 식 (11)에 따라 $f(x_j)$ 를 가진 리더 개체의 위치를 계산하여 업데이트한다. 새로운 위치정보를 가진 리더 개체는 목적함수에 의해 평가된다. 각 개체들의 적합도는 이전 $f(x_i)$ 와 비교되며, $f(x_i)$ 를 재설정한다.

$$U_{goal} = -C_g e^{-\left(\frac{\phi_g}{l_g}\right)^2}, \phi_g = |P_{goal} - P|$$

$$U_{obs} = C_o e^{-\left(\frac{\phi_o}{l_o}\right)^2}, \phi_o = |P_{obs} - P| \quad (23)$$

$$U_{Tot} = U_{goal} + U_{obs}$$

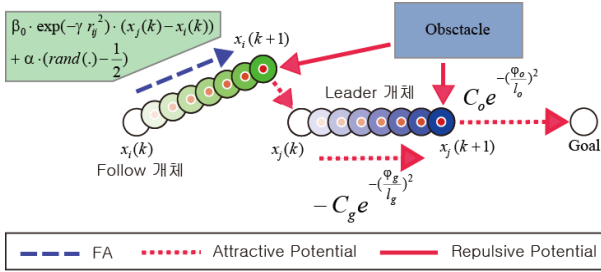


그림 6. 포텐셜 필드를 이용한 위치 업데이트

Fig. 6. Update the position using Potential Fields.

식 (23)에서 $C_{g,o}$ 는 포텐셜의 세기, $l_{g,o}$ 는 포텐셜의 범위, $\phi_{g,o}$ 는 장애물(P_{obs}) 또는 목표점(P_{goal})과 포텐셜을 계산할 점(P) 사이의 거리를 나타낸다[12]. 그림 6은 위치를 업데이트 하는 방식을 표현한 것이다.

Step 7: 식 (24)와 식 (25)에 따라 개체의 이동거리를 제한하고, 식 (26)에 따라 개체간의 충돌을 방지한다.

식 (24)에서 x_{goal} 은 목표점 위치, $x(k)$ 는 현재 개체 위치, x_{move} 는 개체의 x 축 이동, y_{move} 는 개체의 y 축 이동, K_p 는 개체의 제한된 이동거리를 나타낸다.

$$\begin{aligned} & \text{If } |x_{goal} - x(k)| > K_p, \\ & x_{move} = K_p \cos \theta, y_{move} = K_p \sin \theta \end{aligned} \quad (24)$$

식 (25)에서 $x_{move}(k)$ 는 개체의 이동거리를 나타낸다. 그림 7은 개체간의 충돌을 방지하는 방식을 표현한 것이다.

$$\begin{aligned} & \text{If } |x_{goal} - x(k)| \leq K_p, \\ & x_{move}(k) = x_{goal} - x(k) \end{aligned} \quad (25)$$

식 (26)에서 r_d 는 가상 공간의 반지름을 나타낸다.

$$x_i(k+1) = x_i(k) + r_d \frac{(x_i(k) - x_j(k))}{r_0} - (x_i(k) - x_j(k)) \quad (26)$$

여기서 r_d 는 가상공간의 반지름을 나타내고, r_0 는 식 (27)과 같다.

$$r_0 = \|x_i(k) - x_j(k)\| \quad (27)$$

Step 8: 특정 임계값에 도달하고, 모든 개체가 주어진 범위 안에 들어오면 탐색과정을 종료하고, 그렇지 않으면 Step 3부터 반복한다.

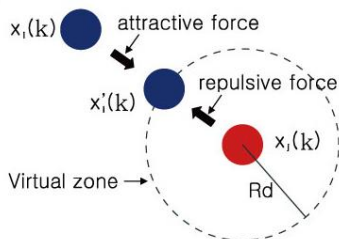


그림 7. 개체 간의 충돌을 방지하는 방법.

Fig. 7. Collision prevention between the agents.

Step 9: 최종적으로 생성된 $x(k)$ 는 군집의 최적의 경로를 가진다.

V. 시뮬레이션

1. 시뮬레이션 개요

본 논문에서 제안하는 FA를 이용한 군집 이동 로봇의 경로계획을 구현하기 위해 Matlab을 이용하여 시뮬레이션을 수행하였다. 우선, 시뮬레이션의 전반적인 프로그램 과정은 그림 8과 같다.

2. FA와 제안된 군집 이동 로봇 경로계획의 비교

기본 FA를 적용한 군집시스템과 제안된 FA를 이용한 군집 이동 로봇의 경로계획에 대한 효율성을 검증하기 위해 개체들이 부피를 가지고 있다는 조건하에 두 가지 방법을 비교하였다. 그림 9와 같이 군집의 개체 이동을 최적화 함으로써 도착 세대 수와 연산시간을 줄일 수 있었다.

3. 개체수 변화에 따른 로봇 군집의 이동

시뮬레이션 대상은 그림 10과 같이 맵 전체 크기가 4m x 3m인 정적인 환경에서 군집의 개체수가 증가함에 따라 군집 시스템이 잘 동작하는 지에 대해 각 스텝당 Trajectory를 표현하였다.

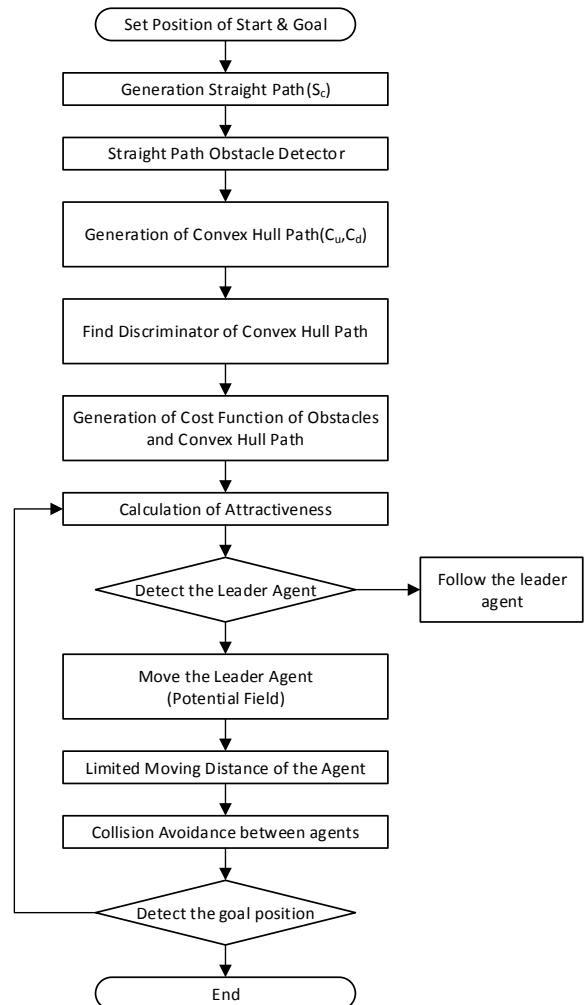


그림 8. 프로그램 흐름도.

Fig. 8. Program Flow-chart.

표 1. 도착 세대수와 연산시간 비교.

Table 1. Generation and computation time.

	FA	New FA
Generation	58	16
Computation time (s)	0.85784	0.48842

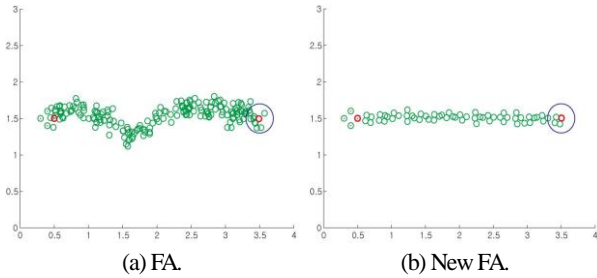


그림 9. 기존 FA와 새로운 FA의 이동 경로 비교.
Fig. 9. Compare with the FA (a) and new FA (b).

그림 9와 같이 개체가 1일 때부터 9까지 늘어날 때 개체 수에 상관없이 군집 시스템이 잘 동작됨을 볼 수 있다.

4. 장애물 회피

시뮬레이션 대상은 그림 11과 같이 맵 전체 크기가 4m x 3m인 정적인 환경에서 다양한 장애물을 가정하여 정의한다. 시뮬레이션 군집의 개체 수는 4개이다. 장애물이 있을 경우, 장애물 회피를 위해 장애물 확장 후 Convex hull를 이용하여 최적 경유점을 추출하였다. 추출된 경유점에 제한한 FA를 이용한 군집 이동 로봇의 경로계획을 이용하여 장애물 회피 시 개체의 각 스텝당 Trajectory를 표현하였다. 그림 10과 같이

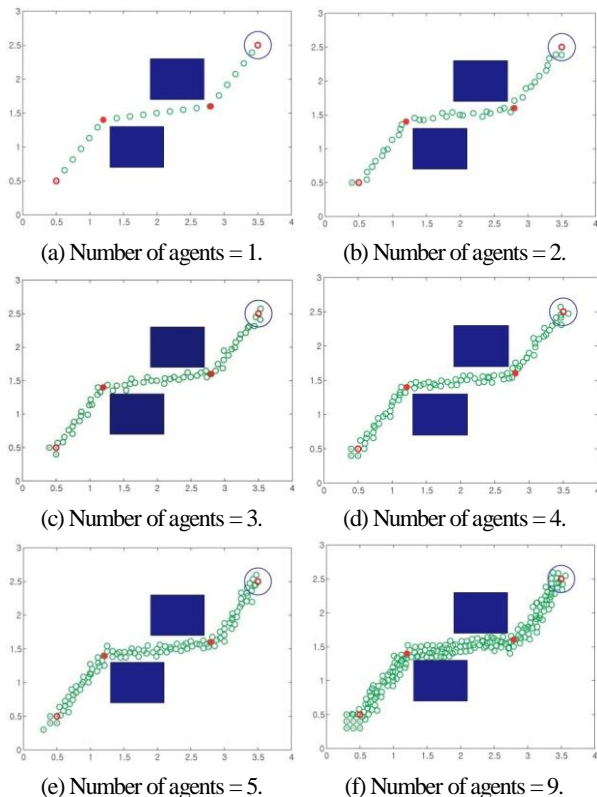


그림 10. 개체수에 따른 장애물 회피 경로.
Fig. 10. Obstacle avoidance according to the number of agents.

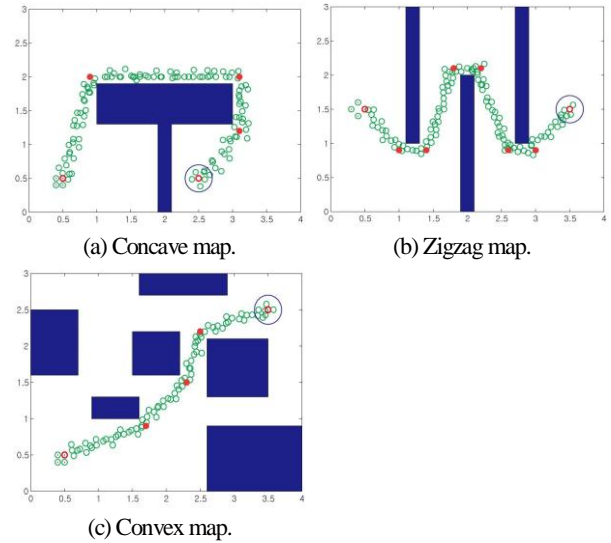


그림 11. 새로운 FA의 다양한 상황에서의 이동 경로.
Fig. 11. New firefly algorithm Trajectory.

장애물이 있는 경우 장애물을 회피하며 군집을 이뤄 목적지 까지 이동함을 볼 수 있다.

5. 군집의 결합 및 분할

각 군집이 군집 결합 없이 이동할 경우 각 군집 간의 개체는 서로를 장애물로 인식하게 되어 각 군집의 이동을 방해하게 된다. 그림 12은 로봇의 개체수가 6일 때 군집 적합도 판정기준에 따라 군집을 나누고, 각 군집의 경유점에 따라 군집의 결합 및 분할을 보여준다.

군집 1은 Start 1→A→B→Goal 1로 이동하며, 군집 2는 Start 2→C→D→E→Goal 2로 이동한다. 군집 3은 Start 2→C→F→G→Goal 3로 이동한다. 이 때 군집 2와 3은 C라는 같은 경유점을 이동하게 됨으로 C까지는 군집 2와 3이 결합하여 하나의 군집으로 이동하며, C 이후에는 각 군집의 목적지에 따라 군집이 분할되어 이동함을 알 수 있다. 군집을 결합 및 분할하는 이유는 FA기반의 군집시스템은 탐색 공간에서 개체 수가 늘어남에 따라 더 많은 정보를 알 수 있으므로 탐색 공간 내에서 더 유연한 움직임을 보여 줄 수 있을 뿐만 아니라 표 2와 같이 도착 세대 수 및 연산시간을 줄일 수 있었다.

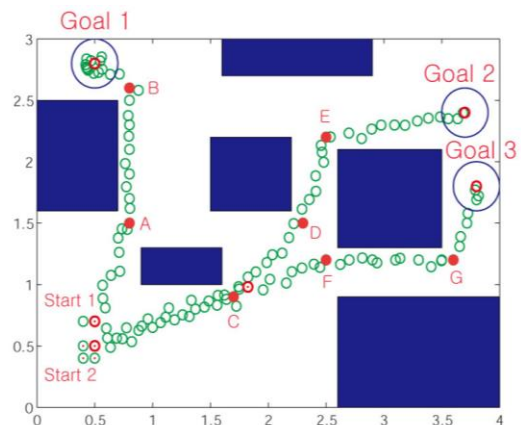


그림 12. 군집 로봇의 결합 및 분할.
Fig. 12. Combined and split of the swarm robots.

표 2. 군집 결합 및 분할을 적용한 결과.

Table 2. Result using combined and split of the swarm robots.

	Without combined and split	With Combined and split
Generation	24	23
Computation Time(s)	12.80413	10.88983

VI. 결론

본 논문에서는 FA를 이용한 군집 이동 로봇의 경로계획을 제안하였다. FA를 이용함으로써 개체 수에 상관없는 군집 시스템을 설계할 수 있었고, Convex Hull Path와 결합함으로써 다양한 장애물을 회피 할 수 있었다. 또 특정한 목적지가 주어졌을 경우 그 목적지를 기반으로 탐색이나 목적지 포위 시스템에 적합한 군집 결합 및 분할 시스템도 제안하였다. 이는 목적함수를 최소화하는 방식으로 움직이는 FA의 개체에서 개체가 늘어남에 따라 Local Minima에 빠질 위험성을 적게 해 로봇 군집 시스템을 더욱 견고하게 하는 역할을 할 뿐만 아니라 서로 다른 목적함수를 발생시킬 필요가 없어 도착 세대 수 및 연산시간을 단축시키는 장점이 있다.

참고문헌

- [1] F. Ducatelle, G. A. Di Caro, and L. M. Gambardella, "Cooperative self-organization in a heterogeneous swarm robotic system," *Proc. of the 12th Annual Conference on Genetic and Evolutionary Computation*, pp. 87-94, Jul. 2010.
- [2] J. S. Kim and Y. H. Joo, "Asynchronous behavior control algorithm of the swarm robot for surrounding intruders," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 9, pp. 812-818, 2012.
- [3] S. J. Russell, P. Norvig, J. F. Canny, J. M. Malik, and D. D. Edwards, "Artificial intelligence: a modern approach," Prentice hall Englewood Cliffs, NJ, pp. 1-60, 1995.
- [4] G Beni, *From Swarm Intelligence to Swarm Robotics*, in *Swarm Robotics*, Springer, Berlin Heidelberg, pp. 1-9, 2005.
- [5] S. Garnier, J. Gautrais, and G. Theraulaz, "The biological principles of swarm intelligence" *Swarm Intelligence*, vol. 1, no. 4, pp. 3-31, Jul. 2007.
- [6] J. H. Lee, J. W. Ahn, and C. W. Ahn, "Energy efficient cooperative foraging swarm robots using adaptive behavioral model," *Journal of Institute of Control, Robotics and Systems (in Korean)*, vol. 18, no. 1, pp. 21-27, 2012.
- [7] X. S. Yang, *Nature-Inspired Metaheuristic Algorithms*, Luniver Pr, pp. 79-90, 2008.
- [8] X. S. Yang, *Firefly Algorithms for Multimodal Optimization*, in *Stochastic Algorithms: Foundations and Applications*, Springer, Berlin Heidelberg, pp. 169-178, 2009.
- [9] G K. Jati, *Evolutionary Discrete Firefly Algorithm for Travelling Salesman Problem*, in *Adaptive and Intelligent Systems*, Springer, Berlin Heidelberg, pp. 393-403, 2011.
- [10] Y. Altintas and K. Erkorkmaz, "Feedrate optimization for spline interpolation in high speed machine tools," *CIRP Annals-Manufacturing Technology*, vol. 52, no. 1, pp. 297-302, Jun. 2003.
- [11] H. T. Kim, *Geometric Path Planning of a Mobile Robot Using B-spline*, Hanyang University (in Korean), Seoul, pp. 5-16, 2009.
- [12] J. O. Kim and P. K. Khosla, "Real-time obstacle avoidance using

harmonic potential functions," *Robotics and Automation, IEEE Transactions*, vol. 8, no. 3, pp. 338-349, Jun. 1992.

- [13] S. Łukasik and S. Żak, "Firefly algorithm for continuous constrained optimization tasks," *Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems*, pp. 97-106, 2009.



김 휴 찬

2011년 한양대학교 자동차공학과 졸업 (공학석사). 2011년~현재 Canon Korea Business Solution 재직중. 관심분야는 군 집 제어 기술.



김 제 석

2011년 한양대학교 자동차공학과 졸업 (공학석사). 2011년~현재 한양대학교 자동차공학과 박사과정에 재학중. 관심분야는 로봇 제어 기술.



지 용 관

2012년 한양대학교 자동차공학과 졸업 (공학박사). 2012년~현재 현대모비스 ADAS설계팀에 재직중. 관심분야는 이 동 로봇 제어 기술.



박 장 현

1983년 서울대학교 기계공학과(공학사), 1985년 서울대학교 기계설계학과(공학석사). 1992년 MIT 기계공학과(공학박사). 1994년~현재 한양대학교 미래자동차공학과 교수. 관심분야는 최적 제어 기술.