

## 유동적인 군집대형을 기반으로 하는 군집로봇의 경로 계획

### An Advanced Path Planning of Clustered Multiple Robots Based on Flexible Formation

위성길<sup>1</sup>, 딜샷사이토프<sup>1</sup>, 최경식<sup>2</sup>, 이석규<sup>1,✉</sup>

Sung Gil Wee<sup>1</sup>, Dilshat Saitov<sup>1</sup>, Kyung Sik Choi<sup>2</sup>, and Suk Gyu Lee<sup>1,✉</sup>

<sup>1</sup> 영남대학교 전기공학과 (Department of Electrical Engineering, Yeungnam Univ.)

<sup>2</sup> 울산마이스터고등학교 전기시스템 제어과 (Department of Electrical System Control, Ulsan Meister High School)

✉ Corresponding author: sglee@ynu.ac.kr, Tel: 053-810-2487

Manuscript received: 2012.7.17 / Accepted: 2012.9.20

*This paper describes an advanced formation algorithm of clustered multiple robots for their navigation using flexible formation method for collision avoidance under static environment like narrow corridors. A group of clustered multiple robots finds the lowest path cost for navigation by changing its formation. The suggested flexible method of formation transforms the basic group of mobile robots into specific form when it is confronted by particular geographic feature. In addition, the proposed method suggests to choose a leader robot of the group for the obstacle avoidance and path planning. Firstly, the group of robots forms basic shapes such as triangle, square, pentagon and etc. depending on number of robots. Secondly, the closest to the target location robot is chosen as a leader robot. The chosen leader robot uses A\* for reaching the goal location. The proposed approach improves autonomous formation characteristics and performance of all system.*

Key Words: Clustered Mobile Robots (군집로봇), Flexible Formation (유동적인 군집대형), Collision Avoidance (충돌회피), Advanced Path Planning (진화된 경로계획)

#### 1. 서론

다수의 로봇이 군집을 이루어 주어진 업무를 효율적으로 수행하기 위해서는 목표점까지 신속히 그리고 정확하게 도달하는 것이 기본적으로 필요하다. 예상치 못한 상황에서 효율적인 진행은 군집 제어를 통해 이루어지며 이는 군집대형, 경로탐색 등과 같이 본 연구를 통해 소개하고자 하는 능동 제어 시스템을 필요로 한다.

최근 인공지능 연구를 통해 로봇 스스로가 생각하고 판단함은 물론이고 군집을 이뤄 유전적 변이에 의해 새로운 사고와 행동의 진화가 일어나 돌발상황을 대처하는 연구가 여러 분야에 걸쳐 진행되고 있다.<sup>1-3</sup> 이전까지의 군집로봇 제어는 모든

행동을 제어하고 통제하는 마스터로봇의 역할이 크다. 이에 따라 마스터 로봇에 다양한 센서 및 고성능의 제어 시스템이 필요하기 때문에 마스터 로봇이 이상을 일으키면 전체 시스템에 큰 손실을 가져왔다. 반면 근래 연구되고 있는 군집로봇은 무리를 짓는 곤충과 새의 무리를 모티브로 하여 그들의 습성, 유전적 변이 연구를 통해 같은 기능을 하는 동종의 이동로봇에서 개체간의 상호작용과 그들만의 행동 습성으로 다양한 행동 양식을 보이며 돌발상황에서 그들만의 즉흥적인 대처 능력을 반복적인 학습을 통해 적응해 나감으로 임무 수행 능력을 향상 시키고 있다. 이런 군집 기술은 탐사, 구조, 물체운반 등 인간을 대신하여 어려운 작업에서 효과적이며, 각각의 로봇이 원하는 경로

로 효과적으로 움직이기 위한 분산 제어 기술, 핵심 센서기술, 로봇간 충돌 방지를 위한 충돌 회피 기술, 경로계획 기술 등 다양한 분야의 기술들이 접목되어 연구되고 있다.<sup>49</sup>

본 논문에서는 군집기술에 필요한 여러 기술 중 장애물을 만났을 때 유동적인 군집대형을 통한 효율적인 회피방법과 그래픽 탐색 알고리즘인 A\* 방법을 적용해 그룹 움직임에서도 최단경로 탐색 방법을 제시한다. 또한 여러 로봇에서의 대형 변화와 각 상황에 적합한 방법들을 제안한다.

**2. 대형제어기술**

군집로봇이란 다수의 로봇이 특정업무 수행에 있어 능동적인 사고를 가지고 서로간의 협업을 통해 효율적인 움직임으로 수행하는 것을 말한다. 이러한 다수의 이동 로봇을 제어하기 위해서는 군집 제어기술을 가지고 임의의 상황에 맞추어 군집 형태를 자율적으로 바꿈으로써 효율적인 움직임을 제어하는 것이 필요하다. 멀티로봇의 초기 군집대형 방식은 로봇의 대수에 따라 Fig. 1 과 같이 기본 형태가 결정된다. 로봇 대수에 따른 대형의 변화는 로봇의 수가 늘어감에 따라 큰 효과를 가져온다. 초기에 로봇이 군집되기 위해서는 기본형태의 대형이 필요하며, 위의 제안은 효율적이고 간단한 방법으로 그 형태를 유지할 수 있다.

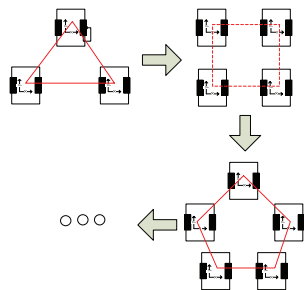


Fig. 1 The basic shapes according to number of robots (triangle, square, pentagon and etc.)

로봇이 기본 군집대형을 갖추기 위한 조건은 아래와 같다.

1. 각각의 로봇간 통신 범위 안에 존재해야 한다.
2. 리더 로봇을 기준좌표로 하여 생성되는 로봇의 좌표들의 최외각 좌표들을 연결하여 그 형태를 결정한다.

3. 형성된 군집대형은 여러 대가 아닌 하나의 로봇으로 인지한다.

군집대형 유지는 다음과 같은 방법으로 진행된다. 리더로봇 좌표가  $(x_n, y_n)$  이고  $n = 1, 2, 3, \dots, \infty$  일 때 삼각대형의 대형유지는 아래와 같다.

If  $n = 3$  ( $n$  은 로봇대수)

$$\text{deviation\_1} = \begin{bmatrix} x_n \\ y_n \end{bmatrix} - \begin{bmatrix} x_{n-1} \\ y_{n-1} \end{bmatrix} \quad (1)$$

$$\text{deviation\_2} = \begin{bmatrix} x_n \\ y_n \end{bmatrix} - \begin{bmatrix} x_{n-2} \\ y_{n-2} \end{bmatrix} \quad (2)$$

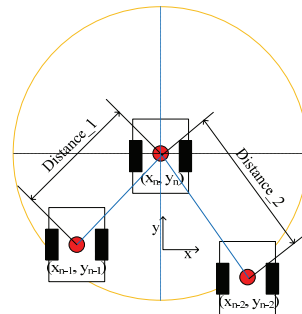


Fig. 2 The detectable range of the clustered robot is selects as a lager circle after comparing with distance between leader robot and others

식(1)과 (2)는 리더 로봇 좌표  $(x_n, y_n)$  에서 두 번째 로봇 좌표  $(x_{n-1}, y_{n-1})$  및 세 번째 로봇 좌표  $(x_{n-2}, y_{n-2})$  사이의 좌표간 편차를 나타내고 있다.<sup>10,11</sup>

또한 리더로봇 좌표를 기준으로 하여 로봇들이 특정 사분면에 위치하게 된다면 자신의 좌표값에 위치된 사분면의 좌표부호를 가지게 된다. 이는 대형유지에서 기준좌표에서 각각의 로봇들이 어느 위치에 존재하는지 정확히 인지하게 된다.

이와 같은 방법으로 로봇의 대수가 증가하면 기준좌표에서 각각 로봇의 좌표거리를 계산하게 되고 (-)부호의 좌표가 나타날 경우 기준 좌표계 원소값에 거리 좌표값만큼 더해줌으로써 정확한 대형이 유지된다. 거리 좌표값의 원소가 (-)부호라는 것은 기준좌표 원소값보다 큰 원소값을 가진다는 의미이다. 따라서 멀티로봇에서 기본 대형유지를 위한 방법은 Fig. 3 에서처럼 좌표계산식이  $n-1$  개이며 식 (1), (2)와 기준 좌표계를 이용해 나타내고 있다.

또한 군집로봇이 목표를 향해 진행할 때 장애물과 조우할 경우, 그룹단위의 회피가 가능하지만 통로를 지나갈 때, 경우에 따라 대형을 변형하여야 하는 경우가 발생한다. 이때 통로에 따른 군집로봇의 대형제어 방식을 정의하고 기술하고자 한다.

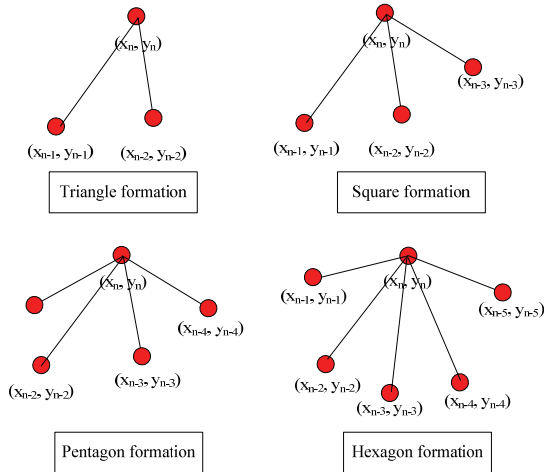


Fig. 3 The basic formations of clustered robots

군집로봇제어에서 대형변형이 필요한 이유는 하나의 그룹으로 인식되는 군집로봇이 장애물 혹은 좁은 통로를 만나게 되었을 때 효율적인 제어를 통한 장애물에 맞는 자동 변형을 하기 위함이다. 군집로봇이 최적의 경로로 나아가기 위해서는 하나의 이동로봇처럼 움직임에 제약이 없어야 하며, 확보된 경로로 나아가야 한다. 이를 위해 군집되어 있던 로봇이 통로의 형태에 따라 그 모습을 최적으로 변형하게 된다.

군집로봇은 세가지 형태의 대형을 취할 수 있으며, 이는 장애물이 없는 지형과 장애물의 형태에 따라 달라짐을 알 수 있다. 첫 번째로 장애물이 없는 공간에서의 군집로봇은 Fig. 3 과 같이 초기 로봇대형이 갖춰진 그대로를 유지하게 된다. 장애물이 없기 때문에 군집로봇을 제어함에 있어 제약이 가해지는 상황이 발생하지 않아 기존의 형태를 그대로 유지하여 목표를 향하는 움직임을 가지게 된다. 그 형태를 유지함에 있어서는 앞에서 언급된 방법에 따르게 된다.

다음으로 장애물의 형태에 따른 군집로봇의 대형 변화는 두 가지 형태로 나타나게 되는데 첫 번째가 Fig. 4 와 같이 수평으로 된 통로와 같은 장애물을 만나게 되었을 때이며, 두 번째로는 Fig. 5 와

같이 수직으로 된 통로 같은 장애물을 만나게 되었을 때 군집로봇이 이를 감지하는 순간 자동적으로 그 형태에 따라 최적의 움직임으로 대형을 형성하게 된다. 이는 Fig. 4 와 Fig. 5 에서 도식화 된 것과 같은 형태를 취함으로써 효율적인 움직임을 보인다.

수직·수평 통로를 군집로봇이 만났을 때 군집 대형의 변형을 통해 회피해가는 순서는 다음과 같다.

1. 군집로봇은 로봇 수에 따른 각각의 고유번호를 가지게 된다. 여기서는 경로 비용이 제일 낮은 로봇이 리더 로봇이다.
  - CN is robot
  - If N = 1, CN is the leader robot
  - If N = 2, CN is the second robot
  - If N = 3, CN is the third robot
  - ⋮
2. 형성된 기본 대형에서 리더 로봇과 가장 큰 편차좌표를 가지는 로봇간의 거리를 계산한다.
  - Leader's coordinate  $(x_n, y_n)$ ,
  - distance\_1  $(x_{dist\_1}, y_{dist\_1})$ ,
  - distance\_2  $(x_{dist\_2}, y_{dist\_2})$
  - If distance\_1 > distance\_2,
  - the radius of clustered robots (R) = distance\_1
  - $R = \sqrt{(x_n + x_{dist\_1})^2 + (y_n + y_{dist\_1})^2}$
  - If distance\_1 < distance\_2,
  - the radius of clustered robots (R) = distance\_2
  - $R = \sqrt{(x_n + x_{dist\_2})^2 + (y_n + y_{dist\_2})^2}$
3. 리더 로봇이 중심이 되고 로봇간 거리 값을 반지름으로 하여 Fig. 4, Fig. 5 와 같은 군집 로봇 범위를 가지게 된다. 이는 로봇이 진행함에 있어 수직 수평 통로와 같은 장애물을 만나게 되었을 때 센서 역할을 하게 된다.
  - Compare distance\_1 with distance\_2 in Fig 2
  - Choose bigger one
4. 통로의 크기와 군집로봇 크기 비교 후 군집 대형 방식을 결정한다.
  - Aisle size (S\_aisle), clustered robot size (S\_cluster)
  - Horizontal aisle in Fig. 4
  - $S_{aisle} < S_{cluster}$
  - Vertical aisle in Fig. 5
  - $S_{aisle} < S_{cluster}$

5. 리더 로봇을 따라 지형지물에 맞춰 Fig. 4, Fig. 5 와 같이 나머지 로봇들이 진행하게 된다.

  - 리더 로봇 좌표,  $(x_k, y_k)$
  - 두 번째 로봇 좌표,  $(x_{k2}, y_{k2})$
  - 세 번째 로봇 좌표,  $(x_{k3}, y_{k3})$

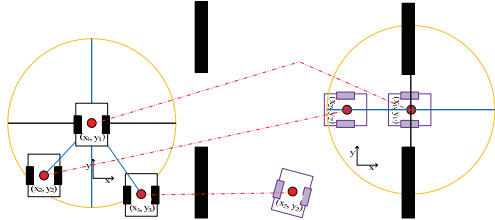


Fig. 4 Formation in case of horizontal aisle

Fig. 4 처럼 수평통로를 지나갈 때, 식(3)-(5)의 과정을 거쳐 진행된다. 식(3)은 시간  $k$  에서의 로봇 3 대의 초기 좌표들을 나타내고 있다. 식(4)는 통로를 인식했을 때 즉,  $k+1$  인 시간에 Fig. 4 의 오른쪽과 같이 변형되는 각 로봇의 좌표이다. 마지막으로 식(5)는 통로를 지나고 난 다음 기본 대형으로 돌아가는 과정의 로봇의 좌표들을 표시하고 있다. 여기서  $k, m$  은 시간 상수이며,  $deviation\_1 = (L_x, L_y)$   $deviation\_2 = (P_x, P_y)$ ,  $k = 1, 2, 3, \dots, m$  으로 나타내어진다.

$$M_k = \begin{bmatrix} x_{k1} & x_{k2} & x_{k3} \\ y_{k1} & y_{k2} & y_{k3} \end{bmatrix} \quad (3)$$

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k1} & x_{k2} \\ y_{k+1} & y_{k1} & y_{k2} \end{bmatrix} \quad (4)$$

$$M_{k=m} = \begin{bmatrix} x_m & x_m - L_x & x_m - P_x \\ y_m & y_m - L_y & y_m - P_y \end{bmatrix} \quad (5)$$

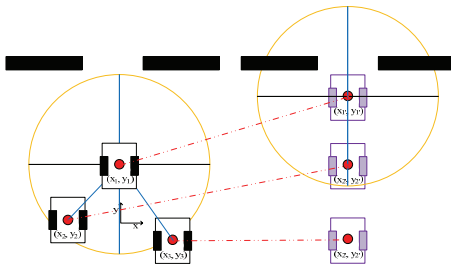


Fig. 5 Formation in case of vertical aisle

Fig. 5 처럼 수직통로를 지나갈 때, 로봇의 움직임 방향에 따른 변형된 로봇의 좌표부분 즉, 식(4)번 부분이 바뀌게 된다. 군집로봇이 오른쪽 방향으로 진행할 때와 왼쪽 방향으로 진행할 때 좌표의  $x$  값을 비교해 고려한다.

오른쪽으로 향하고 있고 리더 로봇의  $x$  좌표 값이 제일 크고 두 번째 로봇, 세 번째 로봇 순일 때 식(6)과 같은 움직임을 나타낸다.

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k1} & x_{k2} \\ y_{k+1} & y_{k1} & y_{k1} \end{bmatrix} \quad (6)$$

$x$  좌표 값이 리더 로봇이 제일 크고 세 번째 로봇, 두 번째 로봇 순일 때 식(7)과 같다.

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k3} & x_{k1} \\ y_{k+1} & y_{k1} & y_{k1} \end{bmatrix} \quad (7)$$

$x$  좌표 값이 두 번째 로봇이 제일 크고 리더 로봇, 세 번째 로봇 순일 때 식(8)과 같다.

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k1} & x_{k3} \\ y_{k+1} & y_{k1} & y_{k1} \end{bmatrix} \quad (8)$$

$x$  좌표 값이 두 번째 로봇이 제일 크고 세 번째 로봇, 리더 로봇 순일 때 식(9)과 같다. 여기서 세 번째 로봇의  $x$  좌표 값은 리더 로봇을 기준으로  $x$  축의 거리가 최지이며 리더 로봇의  $y$  축을 기준으로 대칭이동한 식(10)과 같이 함으로써 대형을 유지한다.

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k1} & x_{k1} - d_1 \\ y_{k+1} & y_{k1} & y_{k1} \end{bmatrix} \quad (9)$$

$$\begin{aligned} x_{k3} - x_{k1} &= d_1 \\ x_{k1} - d_1 \end{aligned} \quad (10)$$

$x$  좌표 값이 세 번째 로봇이 제일 크고 리더 로봇, 두 번째 로봇 순일 때 식(11)과 같다.

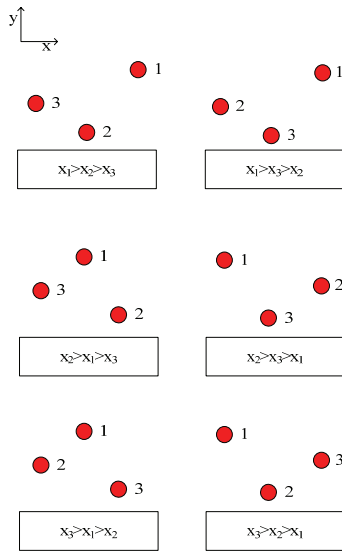
$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k2} & x_{k1} \\ y_{k+1} & y_{k1} & y_{k1} \end{bmatrix} \quad (11)$$

$x$  좌표 값이 세 번째 로봇이 제일 크고 두 번

째 로봇, 리더 로봇 순일 때 식(12), (13)으로 나타낼 수 있다. 식(10)과 같은 방법으로 두 번째 로봇과 리더로봇간 최소 거리값을 나타내므로 두 번째 로봇의 x 좌표값을 리더 로봇의 y 축을 기준으로 대칭이동한 지점으로 보내어 군집대형을 유지한다.

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k_1} - d_2 & x_{k_1} \\ y_{k+1} & y_{k_1} & y_{k_1} \end{bmatrix} \quad (12)$$

$$\begin{aligned} x_{k_2} - x_{k_1} &= d_2 \\ x_{k_1} - d_2 & \end{aligned} \quad (13)$$



$x_1$  : x coordinate of the leader robot  
 $x_2$  : x coordinate of the second robot  
 $x_3$  : x coordinate of the third robot

Fig. 6 Formation changes depending on movement direction of the clustered robots

다음으로 로봇이 왼쪽으로 이동할 때 리더 로봇의 x 좌표 값이 제일 작고 두 번째 로봇, 세 번째 로봇 순일 때 즉, Fig. 6 의 하단 오른쪽과 같을 때 식(6)과 같은 움직임을 나타낸다. Fig 6 의 중간 오른쪽의 형태일 때 리더 로봇의 x 좌표 값이 제일 작고 세 번째 로봇, 두 번째 로봇 순일 때 식(7)과 같게 된다. Fig. 6 의 하단 왼쪽과 중간 왼쪽처럼 대형이 취해졌을 경우 각각 식(8), 식(11)처럼 나타낼 수 있다. 하지만 Fig. 6 의 상단 왼쪽과 상단 오른쪽과 같이 나타내어 질 때 리더 로봇을 기준으로 x 축의 거리가 최저가 되는 로봇을 리더 로봇의

y 축을 기준으로 대칭이동하여 각각 식(14), (15)와 식(16), (17)과 같이 나타내어 군집대형을 변형한다.

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k_1} & x_{k_1} - d_3 \\ y_{k+1} & y_{k_1} & y_{k_1} \end{bmatrix} \quad (14)$$

$$\begin{aligned} x_{k_1} - x_{k_3} &= d_3 \\ x_{k_1} - d_3 & \end{aligned} \quad (15)$$

$$M_{k+1} = \begin{bmatrix} x_{k+1} & x_{k_1} + d_4 & x_{k_1} \\ y_{k+1} & y_{k_1} & y_{k_1} \end{bmatrix} \quad (16)$$

$$\begin{aligned} x_{k_1} - x_{k_2} &= d_4 \\ x_{k_1} - d_4 & \end{aligned} \quad (17)$$

위에서 살펴 본 바와 같이 수직통로의 진입 방향에 따라 군집 로봇이 대형을 변형하여 오른쪽으로 혹은 왼쪽으로 이동해 갈 때 취하는 형태는 Fig. 7 과 같이 됨을 알 수 있다.

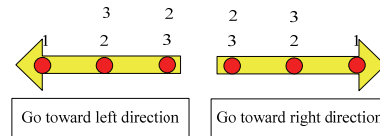


Fig. 7 Formation according to direction

### 3. 그래프 탐색 알고리즘

본 장에서는 군집로봇이 목적지까지 가장 좋은 경로를 찾아가기 위한 경로계획 알고리즘 언급하고 리더로봇 선정에 대해 알아본다. 본 시스템에 적용된 A\* 알고리즘은 그래프 탐색 알고리즘으로 하나의 좌표를 픽셀로 인식하여 시작점에서 목표점까지의 거리비용을 계산하여 최단경로를 수행한다. 이때 군집로봇 내 각각의 로봇들이 목표점까지의 비용을 계산하게 되고 최저 비용을 가지는 로봇이 리더가 되어 최단경로를 수행하게 된다.

시작 노드에서 그 다음 노드로 이동할 때 자신 좌표를 기준으로 한 칸의 거리비용은 1 이 되고 대각 이동 시 1.4 를 가지게 된다. 그 다음 노드로 이동할 때 이전 값과 비교하여 현재점에서 가지는 최저 값을 취하게 되며, 목표 노드까지 도착했을 시 가장 작은 비용 값을 가지는 경로를 선택하게 된다. 한 칸 이동한 비용 값을 1 이라 했을 때 대각

이동 시 피타고라스 정의에 의해 비용은 다음과 같이 계산되며, Fig. 8 과 같이 된다.

$$\sqrt{x^2 + y^2} = \sqrt{2} \cong 1.4 \quad (18)$$

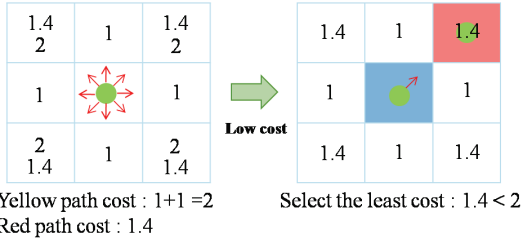


Fig. 8 Decision method of the least cost

가장 잘 알려진 그래프 탐색 알고리즘인 A\* 방법은 Dijkstra<sup>12</sup>와 Greedy Best-First-Search<sup>14</sup>의 장점을 합친 경로 탐색 알고리즘이라 할 수 있다. 본 알고리즘은 목표점 방향으로 진행을 예측(Greedy Best-First-Search)하고 각 좌표점들의 비용 값(Dijkstra)을 구하여 계산량을 줄이고 속도를 향상시켜 최단 경로를 찾는다. 일반적인 표현은  $f(x) = g(x) + h(x)$  이다.<sup>13,14</sup> 여기서  $g(x)$ 는 시작점에서 목표점까지 도달하는 비용 값을 나타내고  $h(x)$ 는 목표점에서 시작점까지 예측된 비용 값을 나타낸다.

Fig. 9 과 같이 장애물이 없는 상황에서의 경로 탐색 과정은 먼저 시작점에서 목표점까지의 최소 비용을 계산하고 즉,  $g(x)$ 를 구하고 난 다음 목표점에서 시작점으로 향하는 동일한 비용 값을 갖는 여러 경로(주황색 경로)를 찾게 된다. 마지막으로

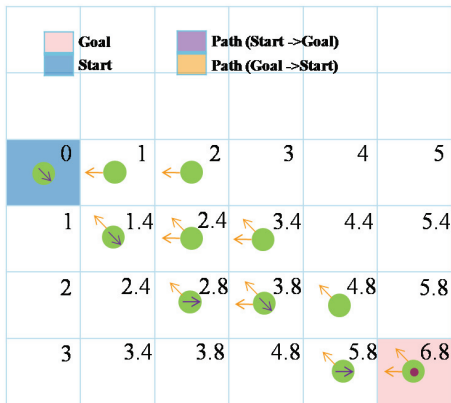


Fig. 9 Finding shortest path for A\* algorithm without obstacles

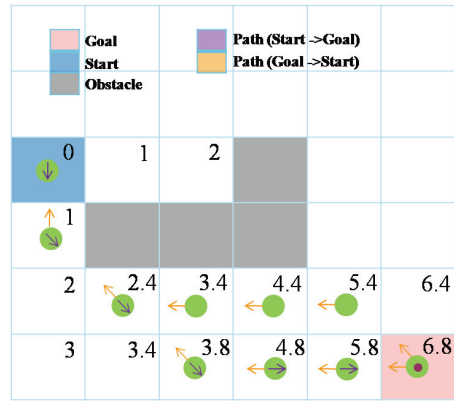


Fig. 10 Finding shortest path for A\* algorithm in obstacle

이들 경로 중 시작점에서 목표점으로 향하는 최단 경로 즉, 비용 값이 큰 쪽을 찾아 목표점으로 가는 경로(보라색 경로)를 선택하게 된다.

Fig. 10 은 장애물이 있는 환경에서의 경로 탐색 과정을 보여주고 있다. 기본원리는 Fig. 9 와 같으며 장애물 부분은 음수값(-1)으로 인식하여 지나갈 수 없는 영역으로 간주하고 알고리즘을 적용하면 주황색 경로는 두 개가 나타나고 보라색 경로 하나가 나타나 최종적으로 Fig. 10 과 같은 최단 경로를 가지게 된다. 이때 Dijkstra 와 Greedy Best-First-Search 알고리즘의 결과와 비교했을 시 계산된 부분과 최단 경로 탐색에 있어 월등함을 알 수 있으며 A\*알고리즘을 적용한 경로 비교 시 현대의 로봇과 초기 좌표값을 가지는 로봇이 리더가 되는 군집로봇에 비해 진화된 군집로봇이 좋은 결과를 가져오는 것을 결과에서 확인할 수 있다.

#### 4. 전체시스템

앞서 살펴본 바와 같이 시스템 구성을 위해 사용한 대형 제어 기술과 최적의 경로 탐색 방식을 알아보았다. 각 부분에 적용된 기술들이 전체 시스템에서 어떤 역할하고 어떻게 사용되었는지 Fig. 11 에서 보여주고 있다.

1. 데이터 파라미터 설정
  - 각 로봇의 위치(시작점), 목표위치(목표점)
  - 맵 전체 정보(장애물, 통로 등)
  - 로봇의 탐색 범위
  - 로봇간 거리에 따른 장애물 가중치

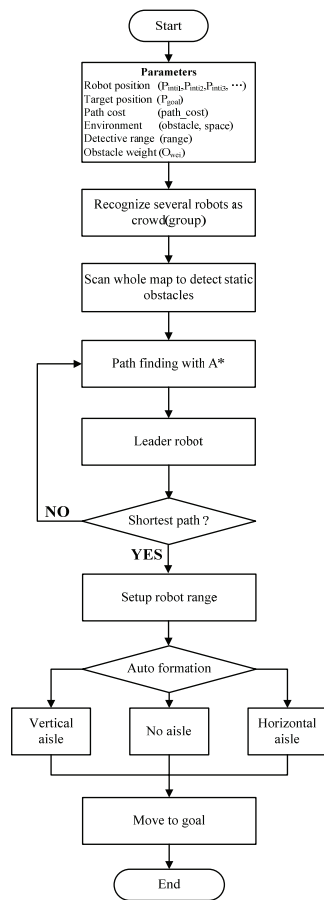


Fig. 11 Flow chart of whole of system

2. 로봇의 대수에 따른 대형 유지 방법
  - 본 논문에서는 3 대의 로봇을 이용한 삼각 대형 방법 적용
  - 대형유지를 통한 군집 인식
3. 맵핑을 통한 고정 장애물의 위치 정보 감지
  - 장애물:  $O_{obs} = -1$
  - 로봇:  $Prob = 1$
  - 빈 공간:  $E_{spc} = 2$
4. 경로 탐색
  - 경로 비용을 통한 리더로봇 선정
  - 그래프 탐색 알고리즘인 A\* 적용
5. 최단경로 유무 판단
  - 경로 비용을 통한 효율적인 경로 탐색
6. 로봇 대형에서 로봇간 거리에 따른 탐지 범위 설정
  - 리더 로봇과 가장 멀리 떨어진 로봇의 거리 값을 이용

7. 유동적인 군집대형 제어
  - 장애물 유무와 형태에 따른 대형 변화 방법
  - 수직·수평 통로에 따른 효과적인 움직임 변화
8. 목표점 도달

### 5. 시뮬레이션 결과

본 장에서는 MATLAB 을 통한 군집로봇의 자동 군집대형 변형과정과 경로탐색 과정의 시뮬레이션 결과를 보여준다. 시뮬레이션에서는 실제환경과 유사한 조건을 주어 테스트 하였으며,  $64 \times 64$  의 맵에서 3 대의 로봇으로 군집을 형성하였다. 여기서 군집로봇의 탐색범위는 앞장에서 설명한 것처럼 리더로봇을 기준으로 하여 가장 먼 거리에 있는 로봇까지의 거리연산을 통해 통로 및 장애물을 감지하게 된다.

군집대형 변형 과정은 3 대의 로봇이 일정 통신 범위 안에서 랜덤하게 놓여 졌을 때 3 개의 좌표가 생기고 이를 그룹으로 인식하여 삼각대형을 구성하게 된다. 그 다음으로 수평·수직 통로를 인식하게 되면 Fig. 12 의 하단과 같이 리더로봇을 기준으로 로봇이 수직·수평대형을 갖추면서 통과하게 되고 3 대의 로봇들이 장애물을 다 통과하게 되면 원래의 초기의 기본 대형으로 돌아가게 된다. 통로가 로봇의 초기 대형보다 작을 시 군집로봇이 자동으로 최적의 상태로 변형하고 통과 후 원래 대형으로 변형한다.

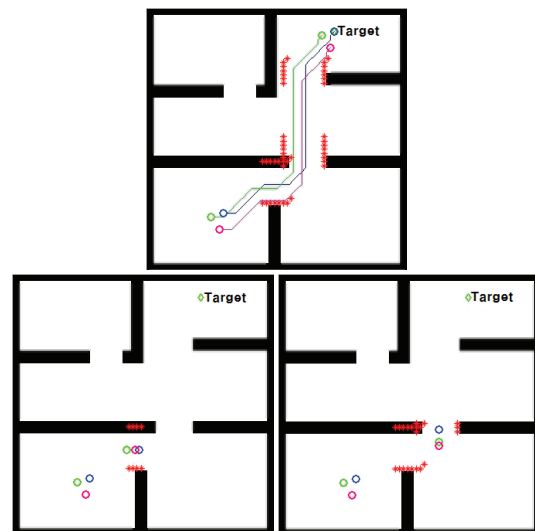


Fig. 12 Simulation results for a clustered robots

Fig. 12 에서 리더로봇은 파란색, 두 번째 로봇은 녹색, 세 번째 로봇은 분홍색으로 표시되며, 빨간색으로 표시된 부분은 군집로봇이 통로를 감지했음을 나타낸다. 또한 수직·수평통로에서의 결과를 바탕으로 실제 환경과 유사한 맵을 구성하여 실험한 결과를 Fig. 12 상단에서 보여주고 있다. Fig. 13 는 Fig. 12 에 대한 시간에 따른 각 로봇과 목표점까지의 거리 비용을 보여주고 있다. 시간 인덱스(Time index) 약 8~15 구간에서는 거의 비슷한 거리 비용을 보이며 이는 Fig. 12 의 하단 왼쪽에서의 수직통로를 통과할 때를 나타냄을 알 수 있다. 수평통로를 통과할 때는 기본대형을 유지할 때와 큰 차이가 없음을 보여주고 있다.

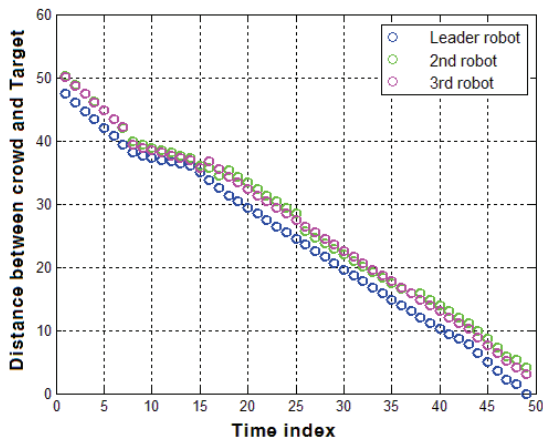


Fig. 13 Measurement of distance to goal

Fig. 14 은 수평·수직통로 뿐만 아니라 장애물을 위치시켜 어떻게 회피해 목표점으로 향하는지를 보여주고 있다. 또한 주목할 만한 점은 Fig. 13 에서 보는 것처럼 장애물의 위치와 크기에 따라 어떠한 상황에서도 자동 군집대형을 통해 최단 경로를 찾아감을 확인할 수 있다. Fig. 13 의 왼쪽은 군집

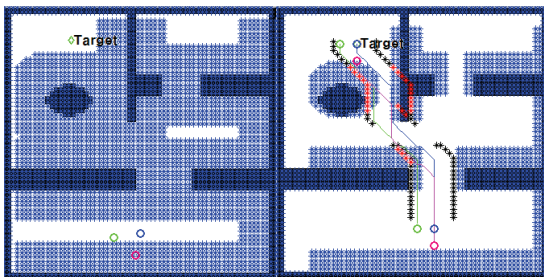


Fig. 14 Simulation results considering obstacle weight

로봇이 그룹으로 인식됨에 따라 로봇의 크기만큼 통로 및 장애물을 지나가야 됨에 따라 장애물의 가중치가 군집의 탐색 범위만큼 주어져 장애물의 각 좌표의 사분면에 적용되어 경로를 찾을 수 없게 된다. 그래서 오른쪽 그림과 같이 장애물의 윗쪽으로 군집로봇의 탐색 범위만큼 가중치를 설정해 줌으로써 최적의 경로를 찾아가게 됨을 보여주고 있다.

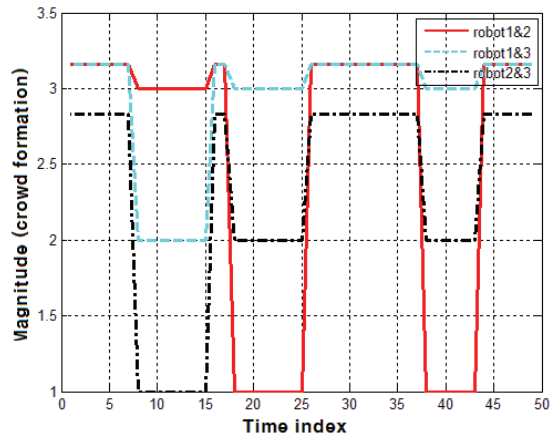


Fig. 15 Comparison of distance of each robot depending on formation

Fig. 15 는 로봇 서로간의 대형유지 상태를 시간별로 나타낸 것이다. 빨간선은 리더로봇과 두 번째 로봇간의 대형유지 상태를 보여주고 있다. 푸른선과 검정선 역시 각각 리더로봇과 세 번째 로봇, 두 번째 로봇과 세 번째 로봇을 나타내고 있으며, 세 번의 대형변형이 시간 인덱스 8~15, 18~25, 38~43 구간에서 일어났음을 알 수 있다.

첫 번째와 세 번째 대형변형구간에서는 첫 번째 로봇과 두 번째 로봇 첫 번째 로봇과 세 번째 로봇간의 변화폭을 발생한 것에서 로봇이 수직 변형이 이루어졌다는 것을 알 수 있다. 또한 두 번째 구간에서는 첫 번째와 세 번째 로봇 그리고 두 번째와 세 번째 로봇간에 변화가 발생한 것으로 보아 군집로봇이 수평으로 대형을 변형시켜 수직 통로를 통과하고 있음을 알 수 있다.

제안된 알고리즘을 적용한 군집로봇으로 Table 1 과 같은 결과를 확인할 수 있다. 장애물이 있는 환경에서 단일로봇이 그래프 탐색 알고리즘으로 목표점까지 이동한 비용과 군집로봇이 이동한 거리가 거의 차이가 없음을 확인할 수 있다. 차이



Table 1 Result of path cost according to types of robots

	맵 (장애물)	단일 로봇	그룹 로봇	군집 로봇
경로	유	51.2132	×	53.799
비용	무	42.8701	42.8701	42.8701

가 나는 이유는 군집로봇이 장애물을 완전히 회피하기 위해 탐색범위를 지정하기 때문에 단일로봇과의 비용 차이를 가진다. 그룹로봇이라 함은 변형이 일어나지 않는 로봇으로 통로보다 그룹이 크다면 통과할 수 없어 경로를 찾지 못하게 된다.

또한 군집로봇에서 업그레이드된 진화된 군집로봇에 제안된 알고리즘을 적용시켜 Fig. 16 과 Table 2, 그리고 Table 3 의 결과를 얻었다. 먼저 Fig. 16 은 목표의 위치에 따른 진화된 군집로봇의 리더 선정 실험을 보여주고 있으며, 그 결과를 Table 2 에서 나타내고 있다. 그리고 Table 3 에서는 장애물이 있을 시 진화된 군집로봇이 가장 좋은 결과를 보이고 있음을 확인할 수 있다.

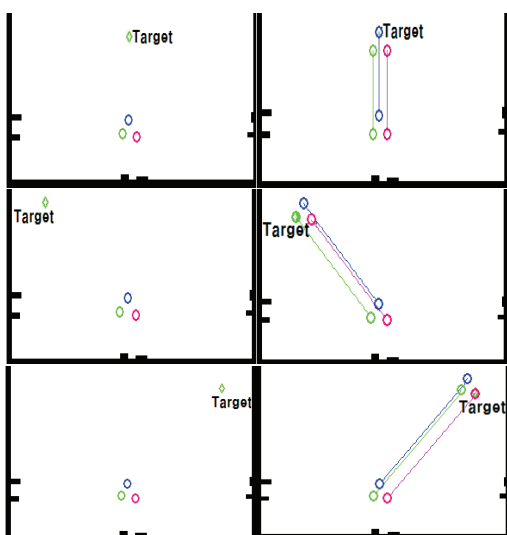


Fig. 16 Decision of the leader robot depending on the goal position

Table 2 The selection of the leader robot without obstacle

경로 비용			리더 로봇
1 <sup>st</sup> 로봇	2 <sup>nd</sup> 로봇	3 <sup>rd</sup> 로봇	
78.6569	85.3137	83	1 <sup>st</sup>
61.2548	58.9117	67.7401	2 <sup>nd</sup>
60.6690	66.7401	59.1543	3 <sup>rd</sup>

Table 3 Comparison of path cost with obstacles

대상	단일 로봇	군집 로봇	진화된 군집로봇
경로 비용	120.6396	120.8112	119.2965

결과적으로 여러 환경과 제안방식을 통해 본 알고리즘을 적용해봄으로써 본 논문에서 제안한 시스템의 효율성을 입증하였으며, 하나의 로봇과 군집로봇이 최단 경로를 탐색하는 것보다 뛰어난 효과를 진화된 군집로봇에서 보여주고 있다.

### 6. 결론

지금까지 제안된 시스템을 통해 진화된 군집로봇의 그룹제어와 상황에 맞는 유동적인 군집대형 변형, 기존의 그래프 탐지 방법들의 장점을 병합한 A\* 알고리즘을 이용해 효율적인 진화된 군집로봇의 리더를 선정하여 능동적인 사고를 가지고 계산된 최단경로 탐색에 대해 알아보았다. 본 연구에서는 로봇 3 대를 이용하였지만 제안된 알고리즘은 다수의 로봇에 대해서도 적용 가능하다. 다만, 로봇의 대수가 늘어남에 따라 처리속도가 늦어지는 부분이 발생하게 될 것이며, 다수 로봇간의 충돌 회피문제를 고려해야 한다.

추후에는 본 연구 결과를 바탕으로 실제 로봇과 유사한 환경을 구성하여, 동적 상황에서도 적용 가능한 알고리즘을 개발하고자 한다.

### 후 기

이 논문은 2012 년도 정부(교육과학기술부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임 (2012R1A1B3002240).

### 참고문헌

- Sharma, Y. K. and Bagla, A., "Security Challenges for Swarm Robotics," International Journal of Information Technology and Knowledge Management, Vol. 2, pp. 45-48, 2009.
- Bang, M. S., Kim, J. S., Joo, Y. H., and Park, J. B., "The Cooperate Navigation for Swarm Robot using Space Partitioning Technique," Proc. of KIEE Summer Conference, pp. 1892-1893, 2011.
- Yoo, Y. D., Jang, S. A., Yang, J. G., Park, J. H., and

- Bae, J.-H. J., "An Implementation of A Multi-Robot System Using Educational Mini-Robots," Proc. of KIISE Conference, Vol. 35, No. 1, pp. 387-390, 2008.
4. Kim, M. K., Ko, K. E., and Sim, K. B., "Behavior Learning and Evolution of Swarm Robot based on Harmony Search Algorithm," Journal of KIIS, Vol. 20, No. 3, pp. 441-446, 2010.
  5. Jung, K. M., Seo, S. W., and Sim, K. B., "Mutual Localization of Swarm Robot using Particle Filter," Proc. of KIIS Spring Conference, Vol. 19, No. 1, pp. 5-8, 2009.
  6. Sim, K. B. and Lee, D. W., "Behavior Learning and Evolution of Individual Robot for Cooperative Behavior of Swarm Robot System," Journal of KIIS, Vol. 16, No. 2, pp. 131-137, 2006.
  7. Kornienko, S., Kornienko, O., Nagarathinam, A., and Levi, P., "From real robot swarm to evolutionary multi-robot organism," IEEE Congress on Evolutionary Computation, pp. 1483-1490, 2007.
  8. Sahin, E., Girgin, S., Bayindir, L., and Turgut, A. E., "Swarm Robotics," Springer-Verlag Verlin Heidelberg, 2008.
  9. Ampatzis, C., Tuci, E., Trianni, V., and Dorigo, M., "Evolution of Signaling in a Multi-Robot System: Categorization and Communication," Adaptive Behavior, Vol. 16, No. 1, pp. 5-26, 2008.
  10. Heo, J. H., Hwang, S. M., Kim, C., and Lee, M. C., "Swarming Robot Control Algorithm Design," Proc. of KIIS Conference, Vol. 20, No. 1, pp. 399-402, 2010.
  11. Das, A. K., Fierro, R., Kumar, V., Ostrowski, J. P., Spletzer, J., and Taylor, C. J., "A Vision-Based Formation Control Framework," IEEE Transactions on Robotics and Automation, Vol. 18, No. 5, pp. 813-825, 2002.
  12. Kang, H., Lee, B., and Jang, W., "Path Planning Algorithm Using the Particle Swarm Optimization and the Improved Dijkstra Algorithm," Proceedings of KFIS Autumn Conference, Vol. 17, No. 2, pp. 176-179, 2007.
  13. Hart, P. E., Nilsson, N. J., and Raphael, B., "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, Vol. 4, No. 2, pp. 100-107, 1968.
  14. Amit Patel, "Pathfinding - introduction to A\*, Heuristics," <http://theory.stanford.edu/~amitp/GameProgramming/>