

# 미지의 환경에서 트리구조를 이용한 군집로봇의 분산 탐색

## Distributed Search of Swarm Robots Using Tree Structure in Unknown Environment

이 기 수\* · 주 영 훈†  
(Gi Su Lee · Young Hoon Joo)

**Abstract** - In this paper, we propose a distributed search of a cluster robot using tree structure in an unknown environment. In the proposed method, the cluster robot divides the unknown environment into 4 regions by using the LRF (Laser Range Finder) sensor information and divides the maximum detection distance into 4 regions, and detects feature points of the obstacle. Also, we define the detected feature points as Voronoi Generators of the Voronoi Diagram and apply the Voronoi diagram. The Voronoi Space, the Voronoi Partition, and the Voronoi Vertex, components of Voronoi, are created. The generated Voronoi partition is the path of the robot. Voronoi vertices are defined as each node and consist of the proposed tree structure. The root of the tree is the starting point, and the node with the least significant bit and no children is the target point. Finally, we demonstrate the superiority of the proposed method through several simulations.

**Key Words** : Swarm robots, Distributed search, Voronoi diagram, Tree structure, Unknown environment

### 1. 서 론

과거 제조용 로봇이 주를 이루던 로봇은 최근 전문 및 개인 서비스용으로 활용분야가 다양하게 응용되면서 제조, 자동차, 의료/재활, 문화, 농업, 국방, 해양, 교육, 연구 등 로봇이 점차 보편화 되고 활용분야가 넓어지고 있다[1]. 과거의 로봇개념이 단순히 자동화(Automation)로 간주되었으나 최근에는 감지(sensing), 사고(thinking & reasoning), 행동(action)이라는 로봇의 3대 키워드가 결합하여 자동화에 자율성이 결합되었다. 로봇이 스스로 자율성을 가지고 로봇이 임무를 수행하려면 주어진 환경 정보의 유/무에 따라 전체적인 시스템 변화와 함께 임무 성공여부도 결정된다. 로봇의 필요성이 강조되는 작업 환경은 사람이 접근하기 불가능하거나 안전을 보장할 수 없는 환경에서의 임무들이다. 재해지역의 인명구조[2], 재난지역의 탐색[3], 외부침입자에 대한 보안감시[4], 행성 탐사[5] 등 제한된 환경에서 인간을 대신하여 탐색과 감시를 하는 목적으로 연구가 진행되고 있다.

집로봇시스템은 단일로봇시스템과 마찬가지로 자율적으로 제어될 수 있으나 다양한 기능적 한계와 환경 상황에 따라 로봇이 효과적인 임무를 수행할 수 없는 상황이 발생한다[6][7]. 단일로봇시스템의 경우 로봇과 센서의 결합이 고장이 발생되면 그 즉시 임무 수행 불가로 이어지며, 군집로봇시스템은 단일로봇시스템에

비해 작업의 효율성과 시스템의 강인함 및 유연성을 확보할 수 있다는 장점을 가진다. 이러한 군집로봇시스템의 특징을 이용하여 군집로봇시스템 핵심 기술의 네 가지 부분으로 군집 행동 제어 기술, 군집 환경/상황인지 기술, 군집 네트워크 기술, 군집 시스템 통합기술로 분류 된다[8]. 본 논문에서는 군집 환경/상황인지 기술인 미지의 환경에서 군집로봇의 탐색에 대해 다룬다.

군집로봇시스템을 이용한 공간을 탐색하는 방법으로 Random 탐색 방식과 Coordinated 탐색 방식으로 나누어진다[9][10]. Random 탐색은 로봇들이 목표를 찾기 위해 탐색하는 동안 탐색 공간을 명시적으로 분할하거나, 동일한 영역을 반복하여 탐색하는 중복탐색에 제한이 없는 방식이다[10]. 로봇의 대수가 증가하여도 전체 시스템이 복잡해지지 않고 제어 시스템을 비교적 쉽게 구축할 수 있는 장점이 있다. 그러나 탐색 완료 소요 시간을 보장 받지 못하며, 각 개체간의 협제 체계를 구축하는 것이 쉽지 않다. Coordinated 탐색은 탐색할 환경을 분할하여 다수의 로봇 대수만큼 균등하게 영역을 나누어 한 로봇 분할영역을 할당하여 탐색을 수행하는 방식이다[11-13]. 로봇의 목표위치를 효율적으로 관리하여 자원 낭비를 줄일 수 있지만 로봇의 대수가 늘어나면 로봇 대수만큼 환경을 다시 분할하여 로봇에게 할당하여야 하므로 제어시스템이 복잡해지고 연산량이 급격하게 증가하는 문제점을 갖는다. 이외에도 로봇이 미지의 환경을 탐색하면서 로봇에 부착되어 있는 센서만으로 외부의 도움 없이 환경에 대한 정확한 지도를 작성하는 작업과 동시에 이동로봇의 현재 시간 동안 자신의 위치를 측정하는 SLAM(Simultaneous Localization And Map-Building)기술에서 사용되는 Wall following 탐색 방식이 있다. Wall following 방식은 일반적으로 벽을 따르거나 객체의 윤곽선을 따르는 것은 실내 구조화 된 환

† Corresponding Author : Dept. of Control and Robotics Engineering, Kunsan National University, Korea.  
E-mail : yhjoo@kunsan.ac.kr

\* Dept. of Control and Robotics Engineering, Kunsan National University, Korea.

Received : December 6, 2017; Accepted : January 30, 2018

경에서 자율 로봇이 장애물을 회피하고 지도를 생성하는데 유용한 탐색 방법이다[14].

분산하는 방법으로 경로를 계획하는 방법이 필요하다. 경로를 계획하는 방법은 크게 전역경로계획과 지역경로계획으로 구성된다. 전역경로계획은 사전에 환경정보를 기반으로 하여 시작점과 목표점까지의 경유점을 결정하여 최적화된 경로를 계획하는 방법이며 최적 경로를 결정하는 많은 연산과 저장 공간이 필요한 단점을 가진다. 지역경로계획은 전역경로계획과 비슷하게 시작위치에서 목표위치까지 경유점을 따라 가지만 동적 장애물을 만나거나 새로운 임무가 할당된 경우 경로를 새로 구성할 수 있는 방법이며 전역경로계획 방법에 비해 적은 연산과 저장 공간 필요한 장점을 가지지만 최적화된 경로를 계획할 수 없고 지역 최소화가 발생할 위험을 포함하고 있다[15]. 경로 계획 방법으로 Genetic Algorithm, 보로노이 다이어그램, 가시성이진 트리 알고리즘 등이 있다. 이러한 알고리즘은 탐색을 통해 얻어진 환경정보를 통해 시작위치에서 목표위치까지 장애물과 충돌하지 않고 지정된 경유점을 이용하여 목표 위치까지 이동하는 방법이다. 지역최소화 현상과 로봇 개체 수 증가에 따라 로봇이 분산되는 방법이 필요하다[15-19].

본 논문에서는 미지의 환경에서 트리구조를 이용한 군집로봇의 분산 탐색에 대해 제안한다. 제안한 방법은 환경에 대한 사전 정보가 없는 상태에서 군집로봇은 미지의 환경을 LRF(Laser Range Finder)센서 정보를 이용하여 최대탐지거리를 분할하여 4 영역으로 나누고, 분할된 4영역에서 장애물의 위치를 파악하고 장애물의 특징점을 검출한다. 검출된 특징점을 보로노이 다이어그램(Voronoi Diagram)의 보로노이 생성자(Voronoi Generators)로 정의하고 보로노이 다이어그램을 적용한다. 보로노이의 구성요소인 보로노이 영역(Voronoi Space), 보로노이 파티션(Voronoi Partition), 보로노이 버텍스(Voronoi Vertex)가 생성된다. 생성된 보로노이 파티션이 로봇의 경로가 되며, 보로노이 버텍스들은 각 노드라 정의하고 제안한 트리구조로 구성된다. 트리의 루트가 시작점이 되고 최하위버트와 자식이 없는 노드가 목표점이 된다. 군집로봇들을 시작점에서 각 목표점으로 트리구조의 경로대로 분산한다. 마지막으로, 몇 개의 시뮬레이션을 통해 제안한 기법의 응용 가능성을 증명한다.

## 2. 전체 시스템 개요

본 논문은 미지의 환경에서 트리구조를 이용한 군집로봇의 분산탐색을 제안한다. 제안하는 기법은 로봇의 탐색과 트리구조를 이용한 군집로봇의 분산으로 구성된다. 로봇의 탐색은 미지의 환경에서 로봇의 전방을 탐지하는 LRF센서를 통해서 탐색방향을 결정한다. 이때, 장애물을 탐지하고 로봇간의 충돌을 회피한다. 군집로봇의 분산은 탐지한 장애물의 특징점을 보로노이 다이어그램을 적용하여 보로노이 버텍스를 추출하고, 추출된 보로노이 버텍스를 각 노드라 정의하고 트리구조로 만들어 로봇들을 분산하여 탐색한다.

그림 1은 전체 시스템 구성도이다. 먼저, 이동로봇의 양쪽 바

퀴에 엔코더가 장착되어 있어 모터의 회전속도 측정하여 로봇의 위치와 회전 각도를 계산한다. 엔코더 정보와 전방의 탐지범위를 가지는 LRF센서의 정보를 통해 로봇 자신의 위치를 추정하고, 장애물을 탐지하여 탐색방향을 설정한다. 탐지된 장애물의 특징점을 추출하고, 추출된 특징점을 보로노이 다이어그램의 생성자 적용하여 보로노이 버텍스를 검출한다. 검출된 보로노이 버텍스는 각 노드라 정의하고 트리구조로 구성하여 로봇들을 분산시키고 탐색한다. 이때 로봇들은 동일한 성능을 가진다.

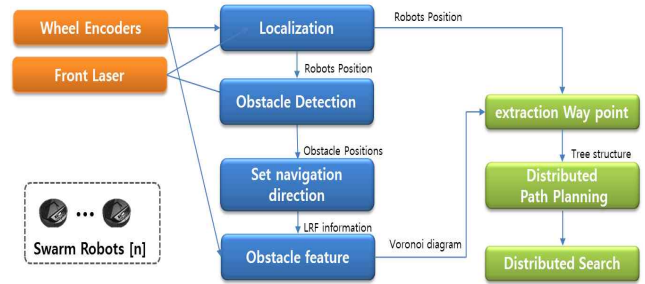


그림 1 전체 시스템 블럭도  
Fig. 1 System block diagram

## 3. 미지의 환경에서 로봇의 탐색

### 3.1 이동로봇의 모델링

이동로봇은 주어진 환경에서 안내, 주변감시, 물건전달 등의 미션을 올바르게 수행하기 위해서는 자신의 위치를 추정해야한다 [5]. 본 논문에서는 그림 2와 같이 독립적 제어 방식인 이륜구동 방식의 이동 로봇을 이용하고 바퀴의 엔코더를 이용한 Dead Reckoning 방법을 사용한다. 그림 1과 같이 로봇이 있을 때, 바퀴간의 거리는  $D$ , 바퀴의 반지름은  $r$ 이라 한다. 로봇의 선속도 (Linear velocity)  $v$ 와 각속도(angular velocity)  $\omega$ 을 아래의 식을 통해 수할 수 있다. 그리고 최종적으로 Runge-Kutta 공식을 이용하여 이동한 위치와 각도의 근사 값을 구할 수 있다.

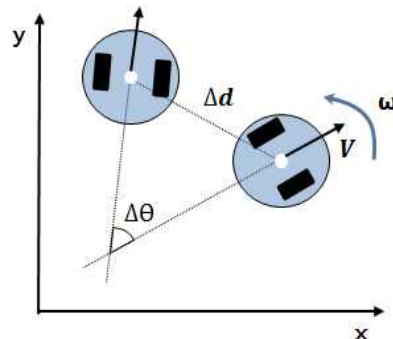


그림 2 로봇의 모델링  
Fig. 2 Robot modeling

$$v_l = \frac{E_{lc} - E_{lp}}{T} \frac{\pi}{180} \quad (\text{rad/s}) \quad (1)$$

식 (1)에서  $V_l$ 는 좌측 휠의 각속도이며,  $E_{lc}$ 는 좌측 모터의 현재 엔코더 값이고,  $E_{lp}$ 는 좌측 모터의 이전 엔코더 값을 나타낸다.  $T$ 는 경과 시간이다.

$$v_r = \frac{E_{rc} - E_{rp}}{T} \frac{\pi}{180} \quad (\text{rad/s}) \quad (2)$$

식 (2)에서  $V_r$ 는 좌측 휠의 각속도이며,  $E_{rc}$ 는 우측 모터의 현재 엔코더 값이고,  $E_{rp}$ 는 우측 모터의 이전 엔코더 값을 나타낸다.

$$V_l = v_l r \quad (\text{m/s}) \quad (3)$$

$$V_r = v_r r \quad (\text{m/s}) \quad (4)$$

$$v = \frac{V_r + V_l}{2} \quad (\text{m/s}) \quad (5)$$

$$\omega = \frac{V_r - V_l}{D} \quad (\text{rad/s}) \quad (6)$$

여기서,  $V_l$ 는 좌측 휠의 선속도,  $V_r$ 는 우측 휠의 선속도,  $v$ 는 로봇의 선속도,  $\omega$ 는 로봇의 각속도를 나타낸다.

선속도 값과 각속도 값을 Runge-Kutta 공식을 이용하여 이동한 위치의 근사 값  $x$ ,  $y$ 와 회전 각도  $\theta$ 를 추정한다.

$$x_{k+1} = x_k + T_c v_k \cos(\theta_k + \frac{T_c \omega_k}{2}) \quad (7)$$

$$y_{k+1} = y_k + T_c v_k \sin(\theta_k + \frac{T_c \omega_k}{2}) \quad (8)$$

$$\theta_{k+1} = \theta_k + \omega_k T \quad (9)$$

본 연구에서는 2개의 모터로부터 얻어진 엔코더 정보를 분석하여 로봇의 움직임을 예측한다. 이 경우 로봇의 불확실성과 바퀴의 미끄럼과 오차로 인해 실제 사용가능한 위치 추정이 불가능하지만 본 논문에서는 오차가 포함되지 않는 정확한 값이라고 가정한다.

### 3.2 LRF센서를 이용한 이동로봇의 탐색 방법

미지의 환경이라도 로봇이 분산하여 탐색하려면 환경 정보가 필요하다. 본 연구에서는 미지의 환경을 탐색하는 방법으로 적합한 Random 탐색과 Wall Follow 주행 방법을 병행하여 사용하는 방법을 제안한다.

그림 3은 본 논문에서 제안한 로봇의 탐색방향을 설정하기 위한 분할영역을 나타낸다. 그림에 의하면 로봇의 모든 LRF 데이터를 검사하지 않고, 로봇의 LRF 센서 전방 영역을 4영역으로 분할하여 장애물의 위치를 판별하고 탐색방향을 설정한다. 4영역은 로봇의 왼쪽(Left), 전방 왼쪽,(Front Left), 전방 오른쪽(Front

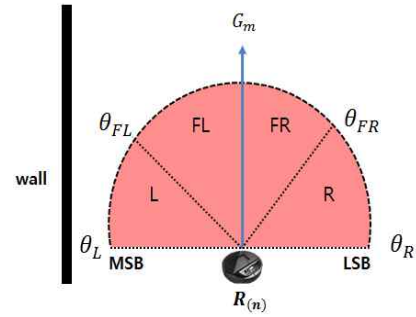


그림 3 로봇의 탐색방향을 위한 분할영역  
Fig. 3 Section aera for search direction of robot

Light), 오른쪽(Light) 영역이다. 영역을 구분 짓는 각도는  $\theta_R$ ,  $\theta_{FR}$ ,  $\theta_{FL}$ ,  $\theta_L$ 으로 설정하였고, 각 영역에 LRF 최대 탐지거리에 장애물이 감지되면 장애물과의 카운트를 통해 장애물이라 판단한다. 장애물의 상태는 분할된 4영역에 장애물이 존재하는지에 따라 분할한다. 여기서 장애물 판단 거리 밖의 센서 데이터에 대해서는 센서의 Max\_range를 대입한다.

표 1은 LRF센서 분할영역의 장애물 분포와 탐색방향 나타낸다. OBS\_bits는 분할영역 내의 장애물의 위치를 나타낸다. 여기서 CLEAR는 전방에 장애물이 없음, L은 왼쪽 장애물, R은 오른쪽 장애물, F는 전방 장애물, ALL은 전방에 장애물이 모두 존재하는 것을 나타낸다. 장애물 분포는 분할영역 L, FL, FR, R를 MSB(Most Significant Bit)를 L, FL 구성하고, LSB(Least Significant Bit)를 FR, R로 구성하여 장애물이 존재하는 영역에는 1, 장애물이 없는 영역은 0으로 표시하였다. 탐색 방향은 장애물 분포에 따라 다음 탐색 방향을 나타낸다. F는 전방의 방향, R은 오른쪽 방향, L은 왼쪽 방향을 말한다. LRF센서의 영역을 4개의 영역으로 분할하여 장애물을 판별하고 장애물의 위치를 알아냈다. 장애물의 위치를 통해 탐색 방향을 설정하였다. Random 탐색 방식과 Wall following 탐색 방식의 병행 방법은 다음과 같다.

표 1 분할영역의 장애물 분포 분석과 탐색방향 설정  
Table 1 Analysis of obstacle distribution and direction setting of partition aera

OBS_bits	분할영역의 장애물 분포	탐색 방향
CLEAR	0000	F
L	1000	R
R	0001	L
L_R	1001	F
FL	0100	R
FR	0010	L
F	0110	F or L
L_FL	1100	L
FR_R	0011	R
L_F	1010 or 1110	R
F_R	0101 or 0111	L
ALL	1111	R

- 1) 일반 주행중 전방 4영역 중 장애물을 만나면 감속하며 접근한다.
- 2) 접근중에 로봇과 장애물의 거리(obs\_dist)가 되면 회피한다.
- 3) 일반 주행중 좌/우 측면에 장애물 감지되면 설정된 거리(desired\_wf\_dist)를 유지하면서 wall following 한다.
- 4) wall following 중에 설정된 거리(desired\_wf\_dist) 지속적으로 감지가 되지 않으면 일반 주행으로 변경한다.

#### 4. 삼각구조를 이용한 군집로봇의 분산

본 논문에서는 군집로봇의 분산으로 삼각구조를 이용한다. 먼저, LRF센서를 이용하여 장애물의 특징점을 검출하고, 검출된 장애물의 특징점을 보로노이 다이어그램의 생성자로 정의하고 적용하여 보로노이 벡터를 추출한다. 추출한 보로노이 정점을 각 노드로 삼각구조를 구성하여 군집로봇을 각 목표 위치로 분산한다.

##### 4.1 LRF센서를 이용한 장애물의 특징점 검출

미지의 환경에서 장애물의 특징점을 검출하는 방법으로는 로봇의 탐색 방향을 위한 분할영역에서 장애물이라고 판단하고, 탐색 순서에 의해 장애물에 접근한다. 이때 장애물의 레이저 센서 데이터를 분석하여 장애물의 특징점을 검출한다.

그림 4에서  $R_n$ 은 로봇이며,  $V_{(n)}$ ,  $V_{(n+1)}$ ,  $V_{(n+2)}$ 은 탐지된 장애물들의 각각의 특징점을 나타낸다. LRF센서 scan 데이터 값을 그래프로 표현해보면 사각형 형태의 장애물은 V자 형태 분포를 보이며 최소값을 가지는 탐지거리가 사각형 장애물의 꼭지점이자 특징점이 된다. 원 형태의 장애물은 U자의 분포를 보이며 U자의 분포가 시작되는 점이 원형 장애물의 특징점이 된다. 검출된 특징점을 빨간색으로 표시하였고, 검정색은 최대 탐지거리에 장애물이 탐지가 되지 않아 Max\_range 값을 적용하였다. 초록색 점은 센서에 감지된 장애물 영역을 나타낸다.

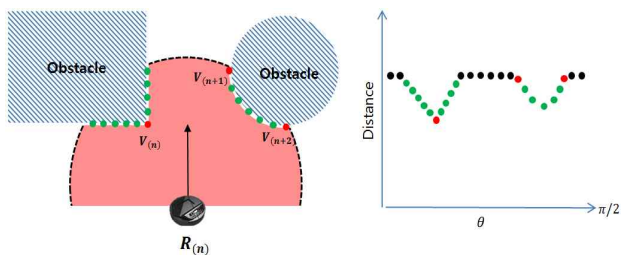


그림 4 LRF센서의 탐지범위  
**Fig. 4** Detection range of LRF sensor

##### 4.2 보로노이 다이어그램을 적용한 경유점 추출

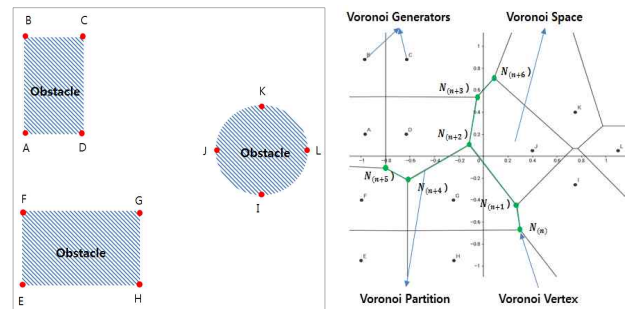
LRF 센서를 이용하여 검출한 장애물들의 특징점을 보로노이 다이어그램을 적용하여 경유점을 추출한다. 보로노이 다이어그램은 평면의 특정 집합의 점까지의 거리에 기초한 평면 영역으로의 분할을 의미한다. 다이어그램은 집합  $S$ 가 있을 때, 한 부분 집합  $S_i$ 에 속하는 원소는 다른 부분 집합  $S_j$ 에 속하지 않고, 집합  $S$ 의

모든 원소는 임의의 한 부분집합  $S_i$ 에 반드시 속하는 성질을 가지는 부분 집합  $S_1, S_2, \dots, S_k$ 을 말한다.

보로노이 다이어그램을 정의하면, 집합  $S$ 의 원소를  $z_i (i=1, 2, \dots, K)$ 라 하자. 집합  $S$ 의 원소  $z, w$ 에 대한 거리함수  $d(z, w)$ 일 때, 보로노이 부분집합  $V_j$ 는 아래 식 (10)과 같다.

$$V_j = \{w \in S \mid d(w, z_j) < d(w, z_i), i = 1, 2, \dots, K, i \neq j\} \quad (10)$$

식 (10)을 만족하는 임의의 한 부분집합  $S_i$ 에 반드시 속하는 성질을 가지는 부분 집합  $S$ 에 속하는 모든 원소의 집합이다. 보로노이 부분집합들의 집합  $\{V_1, V_2, \dots, V_k\}$ 을  $S$ 의 보로노이 다이어그램이라 정의한다. 이때,  $z_i (i = 1, 2, \dots, K)$ 의 집합을 보로노이 생성자라 한다. 보로노이 부분집합  $V_j$ 는 보로노이 영역이라 한다.



(a) 장애물이 있는 환경 (b) 보로노이 다이어그램 적용

그림 5 보로노이 다이어그램 경유점 추출 결과  
**Fig. 5** Extraction result of Voronoi diagram

그림 5는 보로노이 다이어그램을 적용한 경유점 추출의 결과를 나타낸다. 그림 5(a)는 장애물이 있는 환경이며, 빨간색 점이 탐색을 통해 추출된 장애물의 특징점이다. 그림 5(b)는 추출된 장애물의 특징점을 보로노이 생성자(Voronoi Generators)라 정의하고 보로노이 다이어그램을 적용한 결과를 나타낸다. 보로노이 파티션(Voronoi Partition)은 보로노이 영역(Voronoi Space)과 보로노이 정점(Voronoi Vertex) 사이의 선분을 말한다. 이러한 선분들이 만나는 곳이 경유점이며 보로노이 정점이라고 한다.

##### 4.3 삼각구조 구성과 로봇의 경로 생성

본 논문에서는 LRF 센서로 탐지영역에서 장애물의 특징점을 검출하고, 검출한 장애물 특징점을 보로노이의 생성자로 보로노이 다이어그램을 적용하여 경유점을 추출하는 방법을 제안한다. 그림 5에서 보로노이 벡터인  $N_{(n)}, \dots, N_{(n+6)}$ 를 트리의 각 노드라고 정의한다. 트리 구조(Tree Structure)란 자료구조에서 자료의 한 묶음을 노드라고 하는데, 이 노드 사이의 연결을 나뭇가지처럼 생각한 것이 트리구조이다. 트리는 1개 이상의 노드를 갖는 집합으로 노드들은 다음과 같은 조건을 만족해야한다. 트리 T에는 루트(root)라고 부르는 노드가 존재해야하고, T의 나머지 노드들은  $n(\leq 0)$ 개의 분리된 집합  $T_1, T_2, T_3, \dots, T_n$ 으로 분할된다. 여

기서  $T_1, T_2, T_3, \dots, T_n$ 은 트리어이며, 이를 부트리(subtree)라 한다. 한 노드가 최대 두 개의 자식 노드를 가진다. 링크로 연결되어 있는 노드 중에 위에 있는 노드를 부모노드라고 하며, 자식노드는 왼쪽(left)노드와 오른쪽(right)노드이라 한다. 트리의 장점은 노드의 위치를 색인에 의하여 쉽게 접근할 수 있다. 잎(leaf)은 자식을 갖지 않는 노드를 잎 또는 잎노드라고 한다. 방문(visiting)은 노드에 자료값을 읽는 것을 말한다. 그림 5(b)의 경유점인  $W_{(n)}, \dots, W_{(n+8)}$ 까지 각 노드를 그림 6과 같은 형태로 구성하여 나타낸다.

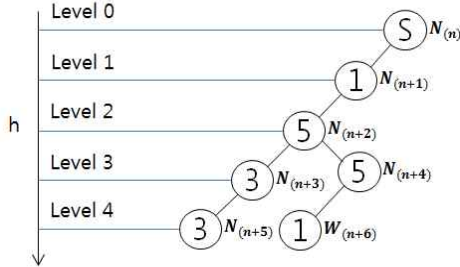


그림 6 경유점 트리구조 구성  
Fig. 6 Tree structure of way point

그림 6에서  $N_{(n)}$ 은 최상위 노드로 루트라 한다. 트리 노드의 개수가  $n$ 개일 때  $n-1$ 개의 간선의 개수를 가지게 된다. 여기서 간선은 부모 노드와 자식노드 사이의 거리를 나타내며 원안의 숫자가 거리의 값을 나타낸다.  $N_{(n)}$ 과  $N_{(n+1)}$ 의 사이 간선은  $N_{(n+1)}$  원안의 숫자가 거리 값 1을 나타낸다. 2차원 평면상의 경유점의 위치를  $x, y$ 좌표 표현하면  $N_{(n)}$ 의 좌표를  $P=(p_x, p_y)$ ,  $N_{(n+1)}$ 의 좌표를  $Q=(q_x, q_y)$ 로 나타낸다. 2차원 좌표상의 거리를 구하는 공식은 유클리디안 거리 식 (11)과 같다

$$\sqrt{(p_x - q_x)^2 + (p_y - q_y)^2} \tag{11}$$

식 (11)은 전체 노드의 부모노드와 자식노드를 계산하여 거리 값을 표시한다. 자식노드가 두 개인 부모노드는 자식노드 왼쪽에 부모노드와 거리가 가까운 값을 가지는 노드로 구성한다. 트리의 높이(height)  $n$ 는 트리가 가질 수 있는 노드의 최소 개수는  $n+1$ 개가 되며, 최대 개수는  $(2^n - 1) + 1$ 개가 된다. 최소 노드 개수는 이진트리 높이  $n$ 가 되려면, 한 레벨에 최소한 한 개의 노드가 있어야하기 때문에  $n+1$ 개가 나오게 된다. 여기서, 최대 노드 개수는 하나의 노드는 최대 2개의 자식 노드를 가질 수 있으므로 레벨  $i$ 에서의 노드 최대 개수는  $2^i$ 개이며 식 (12)와 같다.

$$\sum_{i=0}^n 2^i = 2^{n+1} - 1 \tag{12}$$

여기서,  $n$ 개의 노드를 가지는 트리의 높이의 최대 값은  $n-1$ 개가 되며 최소는  $\log_2 n$ 이 된다. 또한, 루트가 로봇의 시작점이며,

최하위 노드이자 잎노드인  $N_{(n+5)}, N_{(n+6)}$ 이 로봇의 목표위치가 된다. 군집로봇은 공동 경로인  $N_{(n)}, N_{(n+15)}, N_{(n+2)}$ 까지 이동 후에 목표위치인  $N_{(n+5)}, N_{(n+6)}$ 방향으로 각각 분산된다.

5. 시뮬레이션 및 결과 고찰

본 논문에서는 사용된 시스템은 Intel(R) Core(TM) I7 3610QM CPU 2.3Ghz이며, RAM은 8GB, OS는 Ubuntu 16.04 LTS이며 ROS Kinetic버전을 사용하였다. 로봇 실험은 로봇, 센서, 환경 모델 등을 지원하고, 물리 엔진이 탑재되어 실제와 근사한 결과를 얻을 수 있는 3차원 시뮬레이터인 Gazebo로 실험하였다.

본 논문에서는 미지의 환경에서 군집로봇의 효과적인 분산탐색을 연구 목적으로 한다. 본 시뮬레이션에서 입력 파라미터는 표 2와 같고, 주행 환경을 그림 7과 같다.

표 2 시뮬레이션 입력 파라미터  
Table 2 Input parameters for simulation

입력파라미터	명칭	값
S	환경 맵의 크기	20 x 20(m)
n	군집로봇의 개체수	4대
robot_radius	로봇 반지름	0.12
drive_velocity	주행 선 속도	0.3(m/s)
obs_approach	장애물 선행 속도의 접근	0.1(m/s)
obs_avoid	장애물 회피 각속도	0.3(rad/s)
desired_wf_dist	벽면 추종 거리	0.3(mm)
Max_range	LRF 최대탐지거리	300(mm)
Min_range	LRF 최소탐지거리	0(mm)
$\theta_R$	오른쪽 영역 각도	90°
$\theta_{FR}$	전방 오른쪽 영역 각도	40°
$\theta_{FL}$	전방 왼쪽 영역 각도	-40°
$\theta_L$	왼쪽 영역 각도	-90°

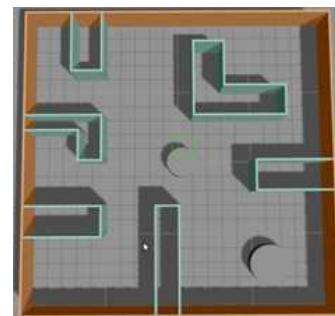
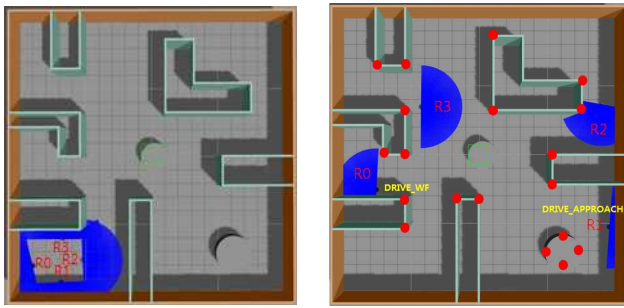
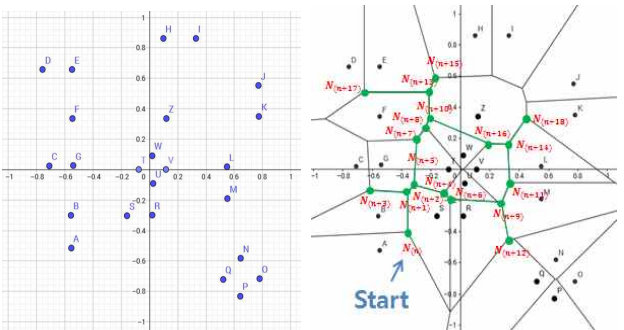


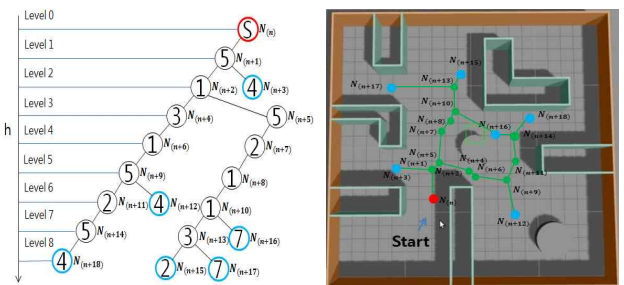
그림 7 시뮬레이션 환경 맵  
Fig. 7 Environment maps for simulation



(a) 로봇들의 초기위치(0sec) (b) 로봇들의 탐색(400sec)



(c) 장애물 특징점 추출 (d) 보로노이 다이어그램 적용



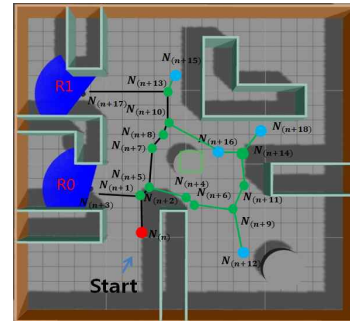
(e) 트리구조 구성 (f) 로봇들의 분산 경로

그림 8 군집로봇들의 경로 생성

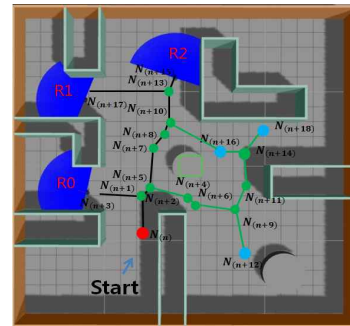
Fig. 8 Path generation of swarm robots

시뮬레이션 환경은 20(m)×20(m)의 공간이며, 초기 탐색은 로봇 4대로 운용하였다. 시뮬레이션 환경에서 로봇의 반지름은 0.12(m)이며, 주행 선속도는 0.3(m/s)이다. 장애물 탐지시 접근 속도는 0.1(m/s)이고, 장애물 회피 각속도는 0.3(rad/s)로 설정하였다. 또한, 탐색방향을 위한 분할영역은 오른쪽, 전방 오른쪽, 전방 왼쪽, 왼쪽으로 각 90°, 40°, -40°, -90°의 각도로 설정하고, 장애물 검출 카운터는 모두 1로 설정하였다.

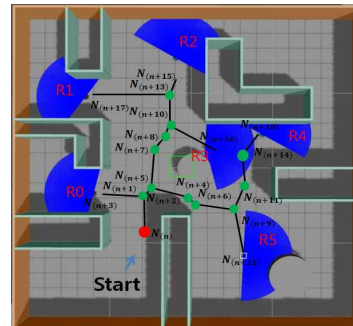
그림 8은 군집로봇들의 경로 생성의 시뮬레이션 결과를 나타낸다. 그림 8(a)는 실험 환경에서 로봇들의 초기위치이며, 그림 8(b)는 400(sec) 후의 로봇들의 모습을 나타낸다. Robot 0은 Wall follow중이며, Robot 2는 장애물에 접근중이다. 여기서, 빨간색 점은 장애물의 특징점이다.



(a) 2대 로봇의 경우



(b) 3대 로봇의 경우



(c) 6개 로봇의 경우

그림 9 군집로봇들의 분산

Fig. 9 Distributed of swarm robots

그림 8(c)는 로봇의 탐색을 통해서 추출한 장애물의 특징점을 좌표 상에 표시한 그림이다. 그림 8(d)는 추출된 장애물의 특징점을 이용하여 보로노이 다이어그램을 적용하여 보로노이 파티션과 보로노이 정점을 추출하는 그림을 보여준다. 그림 8(e)는 보로노이 정점들을 각 노드라 정의하고, 로봇의 초기위치에서 가장 가까운  $N_{(n)}$ 을 최상의 노드로 빨간색 테두리 원형으로 표시하였다. 자식노드가 없으면 하위노드가 로봇들의 목표위치가 되며 하늘색 테두리 원형으로 표시하였다. 목표위치는  $N_{(n+3)}$ ,  $N_{(n+12)}$ ,  $N_{(n+15)}$ ,  $N_{(n+16)}$ ,  $N_{(n+17)}$ ,  $N_{(n+18)}$ 으로 총 6개가 된다. 마지막으로 그림 8(f)는 제안한 방법들을 통해 최종적으로 로봇들의 분산 경로를 생성하였다. 생성된 경로를 군집로봇의 개체수에 따른 군집로봇들의 분산을 나타낸다.

그림 9는 로봇 개체수에 따른 로봇들의 분산을 나타낸다. 그림

9(a)는 로봇 2대로  $N_{(n+3)}$ ,  $N_{(n+17)}$ 을 탐색하고, 다음으로 가까운 거리의  $N_{(n+15)}$ ,  $N_{(n+16)}$  마지막으로  $N_{(n+12)}$ ,  $N_{(n+18)}$ 순으로 탐색한다. 그림 9(b) 로봇 3대로  $N_{(n+3)}$ ,  $N_{(n+17)}$ ,  $N_{(n+15)}$ 을 탐색하고, 마지막으로  $N_{(n+16)}$ ,  $N_{(n+12)}$ ,  $N_{(n+18)}$ 순으로 탐색한다. 그림 9(c) 로봇 6대로  $N_{(n+3)}$ ,  $N_{(n+12)}$ ,  $N_{(n+15)}$ ,  $N_{(n+16)}$ ,  $N_{(n+17)}$ ,  $N_{(n+18)}$ 의 위치로 분산되어 탐색을 실시한다.

## 6. 결 론

본 논문에서는 미지의 환경에서 트리구조를 이용한 군집로봇의 분산 탐색에 대해 제안하였다. 제안한 방법은 환경에 대한 사전 정보가 없는 상태에서 군집로봇은 미지의 환경을 LRF센서 정보를 이용하여 최대탐지거리를 분할하여 4영역으로 나누고, 분할된 4영역에서 장애물의 위치를 파악하고 장애물의 특징점을 검출하는 방법을 제안하였다. 여기서, 검출된 특징점을 보로노이 다이어그램의 보로노이 생성자로 정의하고 보로노이 다이어그램을 적용하였으며, 보로노이의 구성요소인 보로노이 영역, 보로노이 파티션, 보로노이 정점들이 생성하는 방법을 제안하였다. 마지막으로, 몇 개의 시뮬레이션을 통해 제안한 기법의 응용 가능성을 증명한다.

### 감사의 글

본 연구는 2015년도 정부(교육부)의 재원으로 한국 연구재단의 지원을 받아 수행된 연구(과제번호: NRF-2015R1A2A2A05001610)와 2017년도 산업통상자원부의 지역주력 산업육성(R&D) 기술개발 사업의 지원을 받아 수행된 연구임.

### References

- [1] C. Stachniss, "Coordinated multi-robot exploration," *Robotics mapping and exploration*, Vol. 21, No. 3, pp. 43-71, 2009.
- [2] M. A. Subhan, A. S. Bhide, and S. Bhusawal, "Study of unmanned vehicle (Robot) for coal Mines," *International Journal of Innovative Research in Advanced Engineering (IJIRAE)*, Vol. 1, No. 10, pp. 116-120, 2014.
- [3] K. Nagatani, Y. Okada, and N. Tokunaga, "Multirobot exploration for search and rescue missions: A report on map building in RoboCupRescue 2009," *Journal of Field Robotics*, Vol. 28, No. 3, pp. 373-387, 2011.
- [4] P. H. Yuan, K. F. Yang, and W. H. Tsai, "Real-time security monitoring around a video surveillance vehicle with a pair of two-camera omni-imaging devices," *IEEE Transactions on Vehicular Technology*, Vol. 60, No. 8, pp. 3603-3614, 2011.
- [5] M. R. Bouguelia, R. Gonzalez, and K. Iagnemma, "Unsupervised classification of slip events for planetary exploration rovers," *Journal of Terramechanics*, Vol. 73, pp. 95-10, 2017.
- [6] M. A. Neumann, M. H. Chin, and C. A. Kitts, "Object manipulation through explicit force control using cooperative mobile multi-robot systems," *Proceedings of the World Congress on Engineering and Computer Science*, Vol. 1, 2014.
- [7] E. Hernández, b. Antonio, and D. C. Jaime, "Selective smooth fictitious play: An approach-based on game theory for patrolling infrastructures with a multi-robot system," *Expert Systems with Applications*, Vol. 41, No. 6, pp. 2897-2913, 2014.
- [8] S. H. Ji, "Research Trends of Swarm Intelligent Robotics," *The world of Electricity*, Vol. 59, No. 11, pp. 27-31, 2010.
- [9] S. Doctor, G. K. Venaygamoorthy and V. G. Gudise, "Optimal PSO for Collective Robotic Search Applications," *Proceedings of the IEEE Congress on Evolutionary Computation*, Vol. 2, pp. 1390-1395, 2004.
- [10] A. T. Hayes, "How Many Robot? Group Size and Efficiency in Collective Search Tasks," *Proceedings of the International Conference on Distributed Autonomous Robotic Systems*, pp. 289-298, 2002.
- [11] W. Burgard, M. Moors, C. Stachniss and F. schneider, "Coordinated Multi-robot Exploration," *IEEE Transactions on Robotics*, Vol. 21, No. 3, pp. 376-386, 2005.
- [12] J. Cortes, S. Martinez, T. Karata, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, Vol 20, No. 2, pp. 243-255, 2004.
- [13] J. Cortes, S. Martinez, and F. Bullo, "Spatially distributed coverage optimization and control with limited-range interactions," *ESAIM: Control, Optimization and Calculus of Variations*, Vol. 11, No. 4, pp. 691-719, 2005.
- [14] M. Katsev, "Mapping and pursuit-evasion strategies for a simple wall-following robot," *IEEE Transactions on Robotics*, Vol. 27, No. 1, pp. 113-128, 2011.
- [15] M. Wang and J.N.K. Liu, "Fuzzy logic based robot path planning in unknown environment," *Proceedings of Fourth International Conference on Machine Learning and Cybernetics*, Vol. 2, pp. 813-818, 2005.
- [16] G. W. Kim and D. S. Kang, "An Implementation of Object Detection and Tracking Algorithm Using a Fusion Method of SURF and Kalman Filter," *The Journal of Korean Institute of Information Technology*, Vol. 13, No.

2, pp. 59-64, 2015.

- [17] T. Davies and J. Amor, "Multiple waypoint path planning for a mobile robot using genetic algorithms," Computational Intelligence for Measurement Systems and Applications, Proceedings of IEEE International Conference on. IEEE, pp. 21-26, 2006.
- [18] Q. Hui and C. Jin, "Motion planning for AmigoBot with line-segment-based map and Voronoi diagram," Systems Conference (SysCon), Annual IEEE, pp. 1-8, 2016.
- [19] A. T. Rashid, "Path planning with obstacle avoidance based on visibility binary tree algorithm," Robotics and Autonomous Systems. Vol. 61, No. 12, pp. 1440-1449, 2013.

---

## 저 자 소 개



### 이 기 수 (Gi Su Lee)

2016년 군산대학교 제어로봇공학과 졸업.  
 2018년 군산대학교 전자정보공학부 석사과정.  
 <관심 분야> 컴퓨터 비전, 영상처리, 지능형 영상 감시 시스템



### 주 영 훈 (Young Hoon Joo)

1982년, 1984년, 1995년 연세대학교 전기 공학과 졸업 (공학사, 공학석사, 공학박사).  
 1986~1995년 (주)삼성전자 생산기술센터 팀장. 1995년~현재 군산대학교 제어로봇공학과 교수. 1998년~1999년 미국 휴스턴대학 박사 후 과정, 2006년~2007년 제어·로봇시스템 학회 편집주간. 2008년~2009년 한국 지능시스템학회회장. 2016년~현재 군산대학교 풍력기술연구센터장 (교육부선정 중점연구소), 2014년~2017년 International Journal of Control, Automation, and Systems (IJCAS) Editor-in-Chief. 2018년 대한전기학회 차기회장, 평의원. 시니어회원.  
 <관심분야> 지능형로봇, 지능제어, 로봇비전, Human- Robot Interaction, 지능형 감시 시스템, 풍력발전단지제어.