

碩士學位論文

동적 환경에서 분산 로봇  
경로 찾기 알고리즘

指導教授 趙 眞 晟

慶熙大學校 大學院

컴퓨터공학과

金 鶴 壽

2010年 1月 15 日

碩士學位論文

동적 환경에서 분산 로봇  
경로 찾기 알고리즘

指導教授 趙 眞 晟

慶熙大學校 大學院

컴퓨터공학과

金 鶴 壽

2010年 1月 15 日

동적 환경에서 분산 로봇  
경로 찾기 알고리즘

指導教授 趙 眞 晟

이 論文을 工學碩士 學位論文으로 提出함

慶熙大學校 大學院

컴퓨터공학과

金 鶴 壽

2010年 1月 15 日

金鶴壽의 工學碩士 學位論文을 認准함

主審教授 洪 忠 善 (印)

副審教授 李 聖 元 (印)

副審教授 趙 眞 晟 (印)

慶熙大學校 大學院

2010年 1月 15 日

國文要約

## 동적 환경에서 분산 로봇 경로 찾기 알고리즘

### A Decentralized Robot Navigation Algorithm in Dynamic Environments

慶熙大學校大學院

컴퓨터공학과

金鶴壽

로봇공학은 한 로봇만이 아닌 여러 로봇과 장치들이 협력, 협동을 통하여 한계 이상의 일을 처리하는 형태로 발전하고 있다. 이러한 협력과 협동을 위해서는 정보 및 명령을 주고받는 통신능력과 위치 이동 능력이 필요하다. 본 연구는 이러한 많은 수의 로봇으로 이루어진 Networked Robots 시스템에서 발생한 정보를 처리하고, 명령을 내려 목적지까지 이동하는 분산 로봇 경로 찾기 알고리즘을 제안한다. 모든 정보를 이용하는 중앙 집중 방식은 최적경로를 찾을 수 있지만 두 가지 문제를 가지고 있다. 첫 번째, 확장성이 떨어

지게 된다. 한 개의 로봇이 목적지까지 경로를 계산함으로 인하여 많은 계산량을 가지게 되며, 환경이 커짐에 따라 경우 계산량은 계속 증가한다. 두 번째, 통신 사용량이 증가한다. 국부적인 환경 변화에 시작로봇까지 정보를 전송하기 위해 메시지를 전송함으로서 전체적인 통신 사용량이 증가한다. 본 논문은 동적으로 변화하는 환경에서 확장성과 낮은 통신 사용량을 가지는 분산로봇 경로 찾기 알고리즘을 제안한다.

키워드 : Robot Navigation, Networked Robots, Distributed Navigation, Voronoi Diagram

## < 목 차 >

1. 서론	1
2. 관련연구	2
2.1 배경연구	2
2.1.1 Networked Robots	2
2.1.2 Robot Navigation	5
2.2 중앙 집중식 경로 설정	6
2.3 보로노이 다이어그램을 이용한 경로 설정	7
3. 제안하는 경로 찾기 알고리즘	8
3.1 Networked Robot 시스템 구성	9
3.2 보로노이 포인트를 이용한 경로 설정	11
3.3 분산 로봇 경로 설정 알고리즘	12
3.4 보로노이 포인트	21
4. 성능평가	23
4.1 시뮬레이션 환경	25
4.2 결과 및 분석	29
5. 결론	34
6. 참고문헌	35

## <그 립 목 차 1>

[그림 1] Networked robots 의 로봇 구성 .....	4
[그림 2] Voronoi Diagram 대상 장애물 지도 .....	8
[그림 3] 보로노이 다이어그램을 수행한 결과 .....	8
[그림 4] 역 이미지화한 지도 .....	8
[그림 5] 최단거리 알고리즘 수행 후 경로 .....	8
[그림 6] N·R의 정보 처리 및 명령 .....	10
[그림 7] N·R의 정보 처리 및 명령 .....	11
[그림 8] Request Message의 전송 .....	13
[그림 9] Centralized 기법이 Ack .....	13
[그림 10] Centralized 기법의 계산 공간 .....	13
[그림 11] Decentralized 기법의 Ack (왼) .....	14
[그림 12] Decentralized 기법의 계산공간 (오른) .....	14
[그림 13] 보로노이 포인트 작성과 전송 .....	15
[그림 14] 보로노이 포인트를 이용한 목적지 로봇까지 메시 지 전송 .....	16
[그림 15] Decentralized 기법의 경로 설정 .....	17
[그림 16] 장애물의 변화에 따른 경로 설정 .....	18
[그림 17] Centralized Repair Message(실선) 과 Decentralized (점선) .....	19
[그림 18] Decentralized 기법의 경로 로봇 탐색 .....	20
[그림 19] 시작단계에서 Centralized 기법의 주변 환경 정보	21
[그림 20] 보로노이 다이어그램을 이용한 보로노이 포인트(1)	22
[그림 21] 보로노이 다이어그램을 이용한 보로노이 포인트(2)	22
[그림 22] Scenario 1의 총 통신량 .....	27

## <그 립 목 차 2>

[그림 23] Scenario 2의 총 통신량 .....	27
[그림 24] Scenario 3의 총 통신량 .....	27
[그림 25] Scenario 4의 총 통신량 .....	28
[그림 26] Scenario 1의 총 계산량 .....	29
[그림 27] Scenario 2의 총 계산량 .....	29
[그림 28] Scenario 3의 총 계산량 .....	29
[그림 29] Scenario 4의 총 계산량 .....	30
[그림 30] Scenario 1의 총 계산 시간 .....	30
[그림 31] Scenario 2의 총 계산 시간 .....	31
[그림 32] Scenario 3의 총 계산 시간 .....	31
[그림 33] Scenario 4의 총 계산 시간 .....	31
[그림 34] Scenario 1의 이동 거리 .....	32
[그림 35] Scenario 2의 이동 거리 .....	32
[그림 36] Scenario 3의 이동 거리 .....	33
[그림 37] Scenario 4의 이동 거리 .....	33

## 1. 서론

현재 로봇을 이용한 응용은 빠르게 발전하고 있다. 로봇공학은 한 개의 로봇이 아닌 여러 로봇이 협력, 협동을 통해 각 로봇의 한계 이상의 일을 할 수 있는 Networked Robots으로 발전하였다. 협력, 협동함으로써 다양한 응용에 유연한 일의 처리가 가능하다. 여러 로봇들이 효과적으로 일을 처리하기 위해서는 정보 교환을 위한 통신 능력과 원하는 위치까지의 경로 설정과 이동능력이 필요하다.

통신을 이용한 로봇 시스템은 인터넷을 통한 로봇의 개발을 통하여 센서네트워크에서 확장된 개념을 통해 유비쿼터스 시스템을 지나 현재 통신을 통해 협력, 협동하는 로봇 시스템의 형태인 Networked Robots으로 발전하였다. 이는 로봇끼리의 통신뿐만 아니라 센서 혹은 다른 장치와 통신할 수 있고, 기존의 통신망을 이용할 수 있다.

이러한 Networked Robots 시스템은 통신을 통하여 정보를 주고 받아 정보의 처리를 통하여 명령을 내리게 되며, 유선 통신을 이용하는 경우와 달리무선 통신을 이용함으로써 자유로운 이동이 가능하다.

로봇은 다양한 어플리케이션이 이용되며 창고, 위험지역에서의 이동로봇의 경우 넓은 지역에 많은 로봇이 분산 배치된다. 로봇은 제한된 주위인식과 통신거리를 가지고 있기 때문에 로봇간 통신을 통해 효율적인 이동이 필요하다.

최근, N·R의 발전에 따라 여러 로봇들이 협력하는 경로 찾기 알

고리즘에 대해 많은 연구가 되었다. 경로 찾기 문제는 공간을 어떻게 공유할 것인가를 해결하는 것으로서 여러 로봇이 최단 거리, 최소 이동시간을 가지기위해 크게 두 가지 문제가 해결되어야 한다. 첫 번째, 여러 로봇이 경로를 설정하는 문제와 두 번째, 다른 로봇과 장애물의 충돌방지와 교착상태를 피해야 한다.

경로를 설정하는 방법을 크게 두 가지로 나눌 수 있다. 모든 정보를 이용하는 중앙 집중 방식은 최적경로를 찾을 수 있지만 두 가지 문제를 가지고 있다. 첫 번째, 확장성이 떨어지게 된다. 한 개의 로봇이 목적지까지 경로를 계산함으로써 인하여 많은 계산량을 가지게 되며, 환경이 커짐에 따라 경우 계산량은 계속 증가한다. 두 번째, 통신 사용량이 증가한다. 국부적인 환경 변화에 시작로봇까지 정보를 전송하기 위해 메시지를 전송함으로써 전체적인 통신 사용량이 증가한다. 본 논문은 모든 환경정보를 받아 경로를 설정하는 중앙 집중 방식의 문제를 해결하기 위하여 분산 경로설정 방식을 제안한다.

## 2. 관련연구

### 2.1 배경연구

#### 2.1.1 Networked Robots

Networked Robots (N·R)[1, 2, 12]이란 네트워크를 통하여 여러 로봇과 센서 및 다른 컴퓨팅장치, 사람과 협력, 협동하는 로봇을 말한다. 다양한 유·무선통신 프로토콜이 사용 가능하고 이러한 통신을 통하여 협력, 협동하여 주어진 일의 처리가 가능하다.

기존의 로봇 시스템은 한 개 로봇의 한계를 가지고 있다. 하지만 N·R은 여러 로봇이 통신을 통하여 협력, 협동하기 때문에 로봇 한 개가 가지는 한계 이상의 일 처리가 가능하고, 더 높은 유연성과 확장성과 안정성을 가진다. 또한 한 개의 로봇에 큰 능력이 필요한 일을 그 보다는 능력이 조금 떨어지지만 협력을 통하여 처리할 수 있으므로 더 작고, 낮은 가격의 로봇으로 같은 일을 처리 할 수 있다.

N·R은 크게 두 분류로 나누어 생각 할 수 있다.

#### **2.1.1.1 Tele-operated**

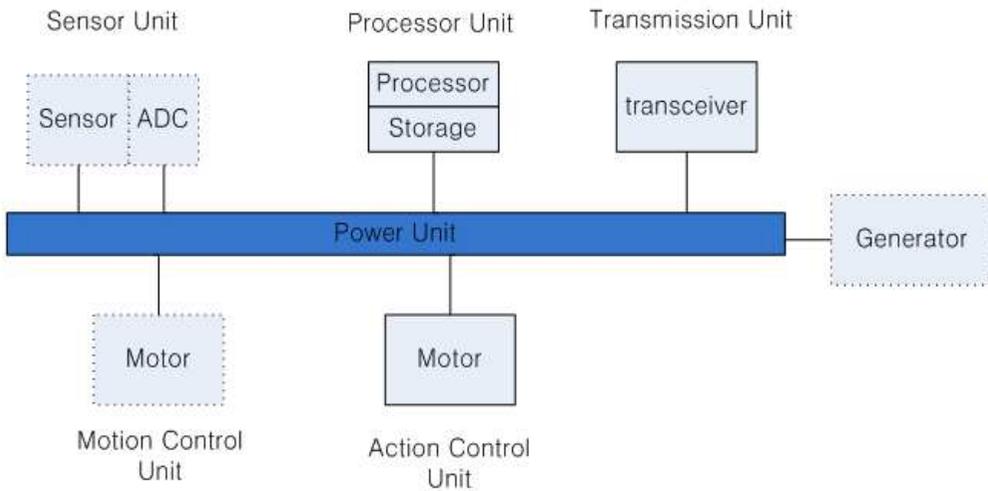
로봇은 통신을 통하여 다른 장치 및 로봇에게 받은 정보 및 명령과 사람으로부터 주어진 명령을 받아 동작한다. 로봇이 자신의 정보만을 이용하는 것이 아니라, 주위의 정보 및 명령을 이용함으로써 효과적인 일의 처리와 협동, 협력이 가능하다.

#### **2.1.1.2 Autonomous**

네트워크로부터 얻은 정보를 이용하여 로봇은 자율적으로 자신의 역할을 수행하게 된다. 기존의 로봇은 연산능력은 없이 주어진 명령만 수행 하게 된다. 하지만 N·R은 주어진 정보를 처리하여 스스로 적합한 자신의 역할을 수행한다.

### 2.1.1.3 Unique Constraints and Challenges

N·R은 로봇공학에 통신이 결합된 형태로, 지속적인 에너지의 공급이 불가능 할 수 있으며, 정보처리 및 명령을 수행하기 위한 계산능력, 저장능력이 필요하다. 그리고 통신장비와 이에 적합한 통신 프로토콜과 이동, 동작을 하기 위한 동력장치가 필요하다. 추가적으로 센서와 구동 이외의 동작을 하는 동력장치가 사용 가능하다.



[그림 1] Networked robots 의 로봇 구성

N·R은 현재 산업현장, 방위산업등 다양한 응용으로 개발되고 있다. 다양한 로봇과 센서와 컴퓨팅 장치로 이루어진 시스템이기 때문에 개발하기 위해서는 다음과 같은 선행되어야 할 과제가 있다.

첫 번째, 협력과 협동을 하기 위한 역할 분담과 협력 계획을 수립하여야 한다. 두 번째, 로봇이 이동하는 경우 다른 환경을 가지게 되므로 유연한 네트워크 환경을 필요하다. 세 번째, 위와 같은 고정

된 형태의 로봇이 아니라면 목적지까지 이동을 위해 위치인식능력이 필요하다. 네 번째, 로봇은 스스로 행동하는 자율성이 필요하며 마지막으로 사람이 N·R의 일부분이 될 수 있도록 사람과 로봇의 협동 및 협력이 필요하다.

#### 2.1.1.4 Applications of N·R

- Intelligent bulidings의 보안

센서나 로봇을 통하여 위협이 되는 상황을 파악할 수 있고, 그 상황에 맞는 반응을 로봇을 통하여 할 수 있다.

- 노인을 위한 하인 로봇

집이나 요양시설에서 N·R은 신체의 불편함을 해결하기 위한 물리적인 일을 대신 할 수 있으며, 기반 시설과 통신하여 긴급 상황 인지 및 대처를 할 수 있다.

- 넓은 지역의 환경 모니터링

기존의 센서네트워크는 고정된 위치에서의 정보를 얻거나 소수의 Actuator로 이루어진 형태로, 넓은 지역의 모니터링에는 한계를 가지고 있다. 하지만 이동이 가능한 여러 로봇을 이용하여 자율적으로 배치, 재구성, 수리능력 통하여 넓은 지역의 모니터링이 가능하다.

- 탐색과 구조작업

이미 많은 로봇이 작은 공간 혹은 위험한 지역의 탐색과 구조작

업에 사용되고 있으며 통신을 통한 협동, 협력을 통하여 긴급하고 위험한 일을 처리할 수 있다.

- 무인 창고

로봇 스스로 물건이 배치되어야 할 장소를 계산 및 명령하여 원하는 위치로 물건을 이동 시킬 수 있다.

- 센서 데이터 수집 [13]

넓은 지역에 적은 숫자만으로 이루어진 지역적인 센서는 에너지와 통신거리의 제약으로 인해 네트워크 형성이 힘들며 목적지까지 데이터를 보내지 못할 수 있거나, 많은 에너지 소모를 통해 전송하게 된다. 이러한 문제를 데이터를 이동 로봇이 근거리로 접근 하였을 때 데이터를 수확하는 방식으로 해결할 수 있다.

## 2.1.2 Robot Navigation

로봇이 목적지까지 이동하기 위해서는 세 가지 단계가 필요하다. 첫째, 전체적, 지역적인 지도를 작성한다. 둘째, 자신의 위치인식을 한다. 셋째, 경로 설정을 한다.

기존의 연구들은 로봇의 이동을 위해 동시 위치지정 및 지도 작성(SLAM: Simultaneous Localization And Mapping)을 이용한다. 이 방법은 이동 로봇이 자신의 위치를 추적하는 동안 국부적인 주변 지역에 대한 지도를 제작하는 것이다. 또한 다중 로봇이 서로 지도 정보를 공유하여 하나의 전체 지도로 결합하는 지도 통합(map merging)을 위한 알고리즘이 개발되었다. 가상의 지도를 쉽게 만드는 것에 비해 로봇이 SLAM을 수행 하기 위해서는 많은 계산과 시간이 소모된다. 이러한 단점을 극복하기 위해 지도 작성을 학습에 의한 어림짐작을 사용하는 방안도 개발 되고 있다.

## 2.2 관련연구

최근, N·R의 발전에 따라 여러 로봇들이 협력하는 경로 찾기 알고리즘에 대해 많은 연구가 되었다.

많은 수의 로봇의 경로를 설정하는 것은 한 개의 로봇의 경로를 설정하는 것 보다 매우 어려운 문제이다. 여러 로봇의 경로 찾기 문제는 공간을 어떻게 공유할 것인가를 해결하는 것이다. 그렇기 때문에 연구된 한 개의 로봇을 위한 경로 찾기 알고리즘은 사용할 수 없다.

여러 로봇이 최단 거리, 최소 이동시간을 가지기 위해 크게 두 가

지 문제가 해결되어야 한다. 첫 번째, 여러 로봇이 경로를 설정하는 문제와 두 번째, 다른 로봇과 장애물의 충돌방지와 교착상태를 피해야 한다.

### 2.2.1 중앙집중식 경로 설정

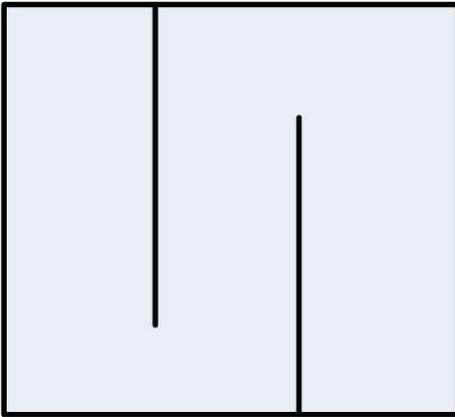
중앙 집중방식은 단 하나의 경로라도 있다면 경로를 찾을 수 있으며 최적 경로를 구할 수 있다. 하지만, 공간이 커지는 것에 급격한 정보량 증가와 이에 따른 많은 계산량 때문에 제한된 시간내 결과를 얻기 위해 기존의 연구는 휴리스틱 기법을 사용하였다.

지금까지 Robot Navigation 정적인 환경에서 주위의 모든 위치정보와 자신의 위치를 알고 있는 상황에서 경로 찾기 알고리즘에 대해 연구되었다. 또한, 여러 로봇으로 이루어진 분산 로봇들의 효과적인 움직임을 위한 알고리즘이 개발되었다.[7, 8, 11, 14] 모든 로봇이 경로 유전자를 가지는[7, 11]기법은 경로가 겹치는 부분을 최소화 하여 충돌과 최단거리 문제를 해결하였다. 우선순위 기반으로 A\*알고리즘을 이용한 기법[8]은 교착상태 문제를 해결하고 경로 찾는 확률을 높였다. 하지만, 로봇은 자신외의 모든 장애물과 다른 로봇의 경로를 알아야 하기 때문에 여전히 많은 통신량과 계산량 문제가 해결되지 않는다.

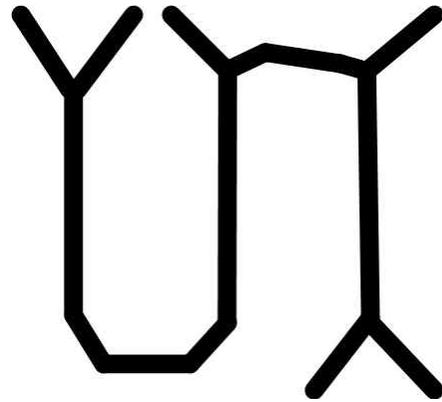
### 2.2.2 보로노이 다이어그램을 이용한 경로설정

보로노이 다이어그램[10]은 이론이 완벽할 뿐 아니라 다양한 응용분야에 연구되어지고 있다. 그중 하나로 길 찾기 알고리즘에도 사용되고 있다. 보로노이 다이어그램에 의해 설정된 경로는 장애물

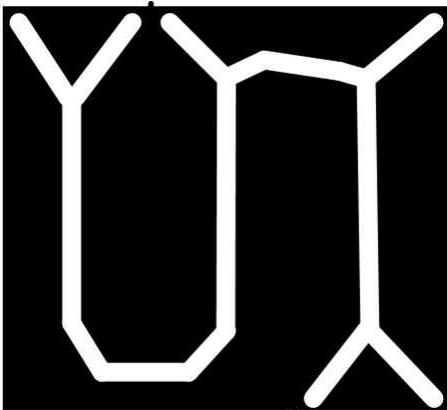
을 최대한 멀리하는 경로로서 충돌을 최소화한 경로이다. [그림2, 3, 4, 5]는 보로노이 다이어그램에 의해 경로가 설정되는 과정이다. [그림3]과 같이 보로노이 다이어그램으로 만들어진 뼈대를 이용하여 역이미지화 한 [그림 4]에서 최단거리 알고리즘을 수행하면 [그림 5]와 같은 경로를 만들 수 있다.



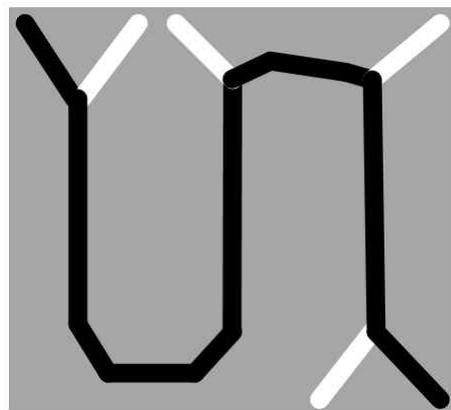
[그림 2] Voronoi Diagram 대상 장애물 지도



[그림 3] 보로노이 다이어그램을 수행한 결과



[그림 4] 역 이미지화한 지도



[그림 5] 최단거리 알고리즘 수행 후 경로

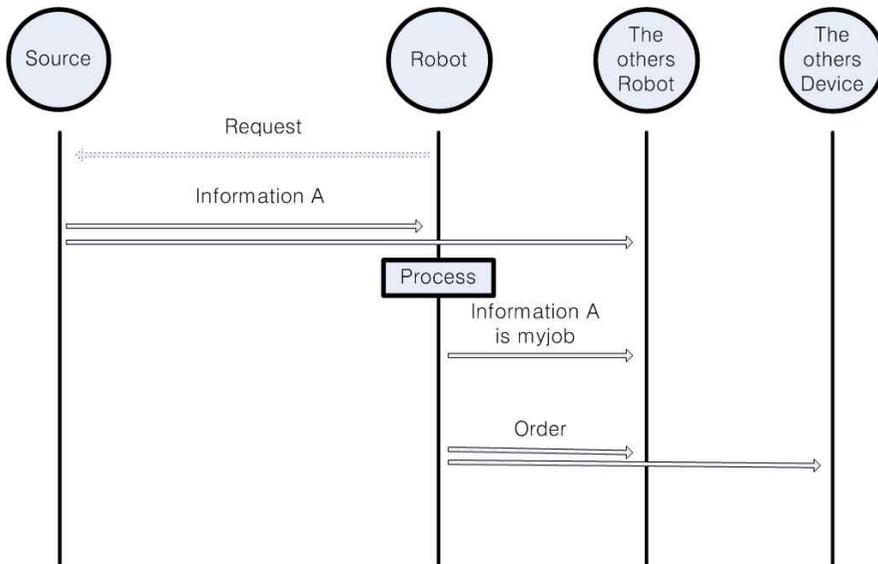
### 3. 제안하는 경로 찾기 알고리즘

넓은 지역에서 많은 로봇간의 충돌방지와 최단경로를 고려한 경로를 만들기 위해서는 많은 양의 정보가 필요하다. 또한, 이 정보를 제한된 시간 내에 알고리즘을 수행 하는 것은 매우 어렵다. 경로를 설정하는 방법은 크게 중앙집중 방식과 분산방식으로 나눌 수 있다. 중앙 집중방식은 최적의 해를 구할 수 있지만 실제 로봇이 이동하기 위해서는 많은 통신과 계산을 제한된 시간에 처리해야 하기 때문에 적합하지 않다. 따라서 분산방식(Decoupled)이 제안되었으나 모든 다른 로봇의 경로와 환경을 알고 있는 상황에서의 경로설정 방법으로 실제 응용에 사용하기 힘들다.

제안하는 방안은 변화하는 환경에서 적은 통신과 계산으로 최적 경로를 설정함으로써 제안된 시간 안에 경로를 구할 수 있으며, 적은 에너지를 소비한다.

#### 3.1 N·R 시스템 구성

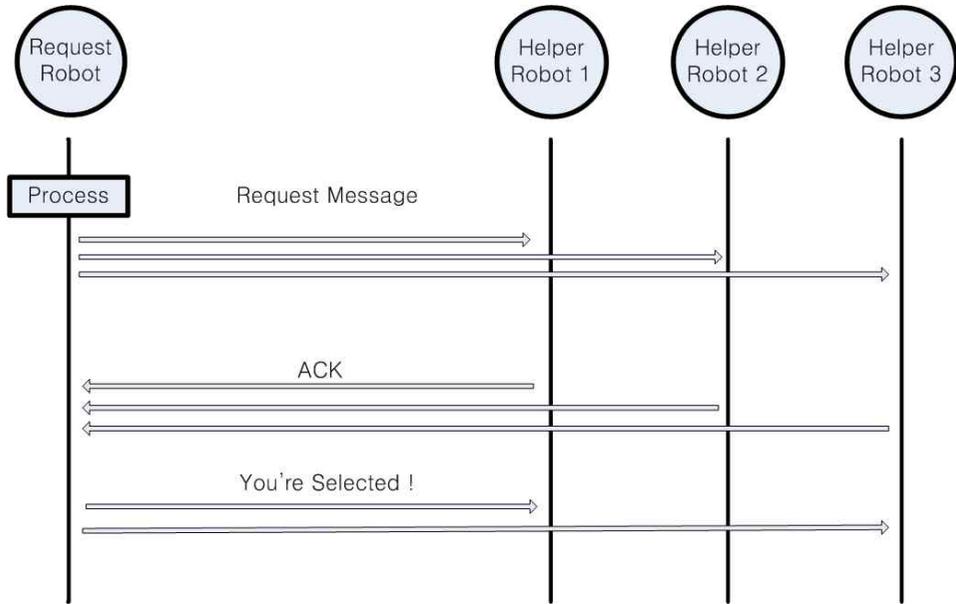
Networked Robots 시스템은 센서 네트워크와 같이 에너지와 계산능력의 제약이 크지 않다. 정보 처리 및 명령을 내릴 수 있는 계산, 통신능력을 가지고 있기 때문에 N·R으로 구성된 시스템은 정보가 Centralized하게 Server에서 처리되어 다시 명령을 내리는 형태가 아니라, Source에서 가까운 로봇이 이를 처리하고 명령을 내리는 Distributed 시스템으로 동작한다. 정보의 처리 및 필요한 로봇 및 장비에게 명령은 다음 [그림 6] 과 같은 형태로 이루어진다.



[그림 6] N·R의 정보 처리 및 명령

정보는 로봇으로부터 요청되어 받을 수 있고, 다른 센서 및 로봇이 전송할 수 있다. 정보는 한 로봇이 처리하며 중복 처리를 막고 병렬적인 계산이 필요한 경우를 위해 다른 로봇에게 계산된 결과 정보를 전송한다.[그림 6] 그 후 필요 로봇 및 장비에게 명령을 내리기 위해 필요 메시지를 Broadcast한다.

로봇은 사용자의 명령 혹은 정보의 처리를 통해 수행해야 할 일이 결정되는 경우에 동작을 시작하게 된다. 사람 혹은 다른 로봇으로부터 명령을 받는 경우에는 바로 동작을 시작하게 되지만, 정보의 경우에는 어떠한 장비가 얼마나, 어디서 필요한지 정보의 처리를 통하여 결정하게 된다.



[그림 7] N·R의 정보 처리 및 명령

로봇은 정보의 처리를 통하여 자신으로부터 가까운 필요한 로봇 및 장비만을 골라 이를 통해 일을 처리한다.[그림 7] 현재 로봇 자신으로부터 어떤 로봇과 장비가 있는지 알 수 없으므로, 요청 메시지를 Broadcast한 후 이에 적합한 장치는 응답 메시지를 보내게 된다. 응답메세지중 가까운 필요 장비에게 선택되었음을 알려 전체적인 일의 시작을 하게 된다.

### 3.2 보로노이 포인트를 이용한 경로설정

보로노이 다이어그램을 이용한 경로 찾기 알고리즘[9]은 충돌방지와 최단거리를 만족하지만 여러 로봇이 이동하는 환경에 적합하지 않다.

보로노이 다이어그램을 수행한 정보는 모든 환경정보를 전송하는

것 보다는 적은 데이터를 보내게 된다. 하지만 선으로 만들어진 경로를 더욱 줄이기 위해 보로노이 다이어그램을 이용한 보로노이 포인트를 제안한다. 로봇은 자신 주변을 센싱한 값을 통해 주위 지도를 만들었을 때 환경 정보를 점으로 전송함으로써 적은 통신량까지 줄 수 있다. 또한 점으로 이루어진 경로설정으로 계산량을 줄일 수 있다.

### 3.3 분산 로봇 경로 설정 알고리즘

위험지역, 경계지역, 무인 창고에서의 자율 이동로봇은 분산 로봇 시스템의 좋은 예이다. 로봇은 주변 환경을 인식하는 센싱 거리의 한계와 통신 거리의 한계를 가지고 이동을 위해 경로를 설정하여야 한다. 넓은 공간에서 많은 수의 로봇이 충돌방지와 최단거리를 고려해야 하며, 제한된 시간내에 경로를 설정해야 한다. 이것은 매우 어려운 일이다.

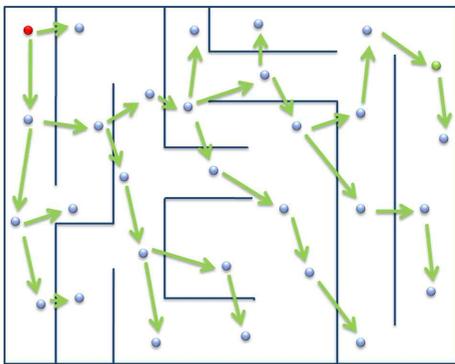
중앙 집중방식은 많은 정보로 인한 통신과 계산의 한계에서 경로를 설정하기 위해 휴리스틱 기법을 사용하였다. 하지만, 여전히 많은 통신량과 한 곳에서의 많은 계산량을 가지며 넓은 공간에서 활용되는 응용에 확장이 불가능하다. 이를 위해 분산 경로설정기법[7, 8]이 제안되었다. 모든 로봇은 평등하게 취급되며 중앙 집중 방식과 같이 통제를 하는 서버나 로봇이 없다. 이러한 기법들은 로봇의 수가 늘어나는 것과 공간이 넓어 지는것에 대해 중앙 집중 방식보다 좋은 결과를 가진다. 하지만, 로봇이 이동할 때 이동 로봇 자신을 기준으로 많은 정보가 집중되며 계산함으로써 인해 제안시간 내에 경로를 만들기 어려운 문제를 가지고 있다.

또한, 분산 경로설정 방법은 분산 시스템이 경로를 찾을 수 있으

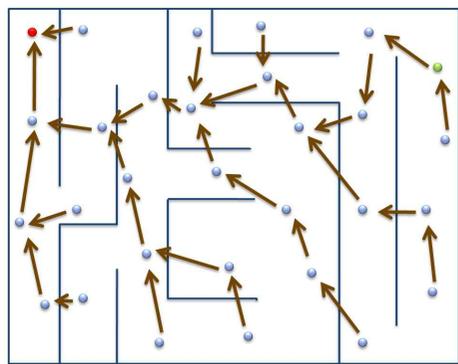
나 출발 로봇을 기준으로 중앙 집중식으로 경로가 설정된다. 한 개의 로봇에 많은 정보가 집중되고 계산하는 문제를 해결하기 위해 본 논문은 Decentralized 기법을 제안한다.

중앙 집중 방식 [그림 8, 9] 와 같이 Supervisor로부터 Request을 보내고 Ack을 받게 된다. 분산 방식도 Source로부터 Request을 보내고 Ack을 받는 방식을 사용한다. 한 개의 로봇에게 많은 데이터가 집중되어 계산되는 방식은 공간이 더 넓어지는 경우 많은 정보를 제한된 시간내에 계산하기 힘들어 진다. 또한 불 필요한 부분의 장애물 지도[그림 10]까지 계산하여 필요하지 않은 계산을 하게 된다.

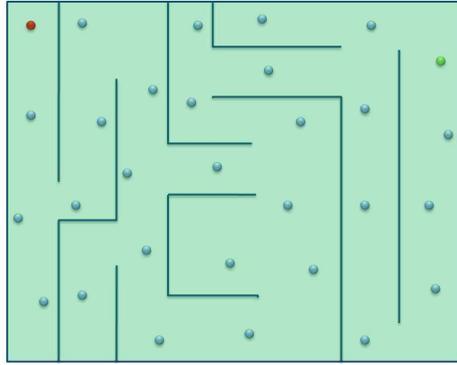
제안된 Decentralized 방식[그림 11] 같이 경로에 필요한 부분[그림 12]의 정보만 받아 분산 계산된다.



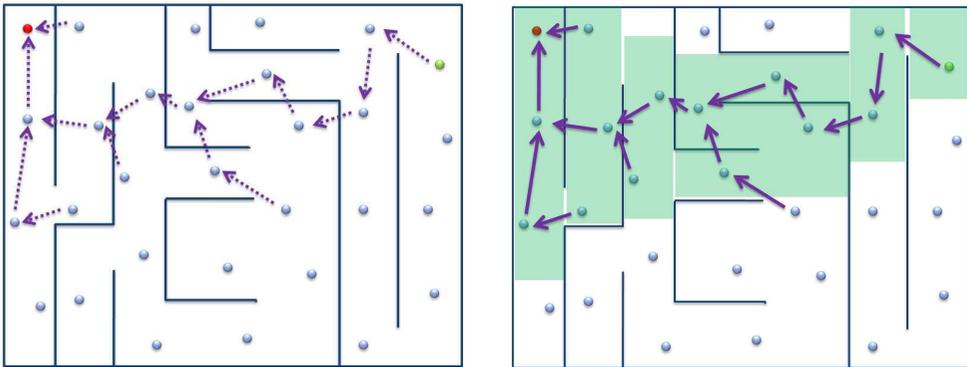
[그림 8] Request Message의 전송



[그림 9] Centralized 기법이 Ack



[그림 10] Centralized 기법의 계산 공간



[그림 11] Decentralized 기법의 Ack (왼)

[그림 12] Decentralized 기법의 계산공간 (오른)

### 3.3.1 Decentralized 기법의 작동 절차

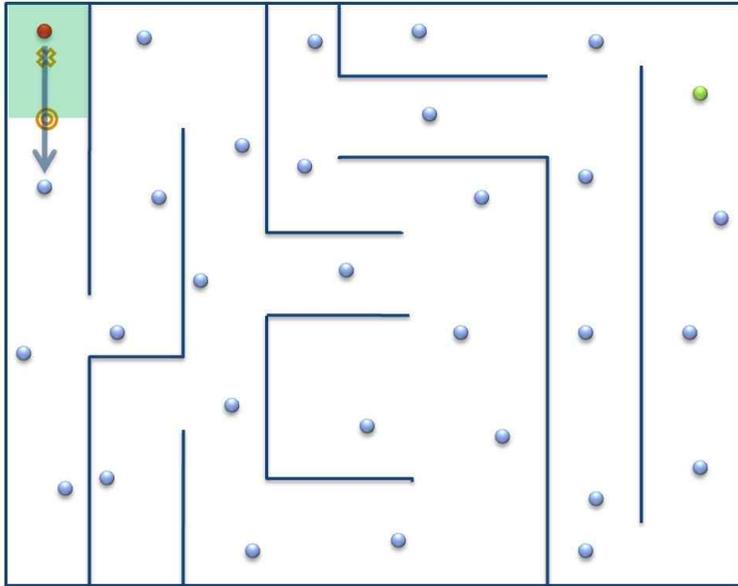
본 논문에서 제안하는 Decentralized 기법은 보로노이 다이어그램을 이용한 보로노이 포인트를 이용하여 적은 데이터 전송과 적은 계산량을 분산하여 수행한다. Request Message 전송과정에서 경로 로봇의 계산을 통해 최단 경로를 설정하게 된다. 주변 환경 탐색 거리보다 통신 거리가 넓기 때문에 경로의 로봇은 자신이 알고 있는 Open Point를 이웃 로봇에게 전송하면 전 로봇이 알고 있는 Open

Point를 알고 있는 로봇에 한해 메시지를 다른 로봇에게 재전송 하게 된다. 이러한 방법을 사용하면 실제 경로가 있는 경로로 메시지가 전송되기 때문에 목적지 로봇은 Source로부터 메시지를 통해 최단경로를 알 수 있다.

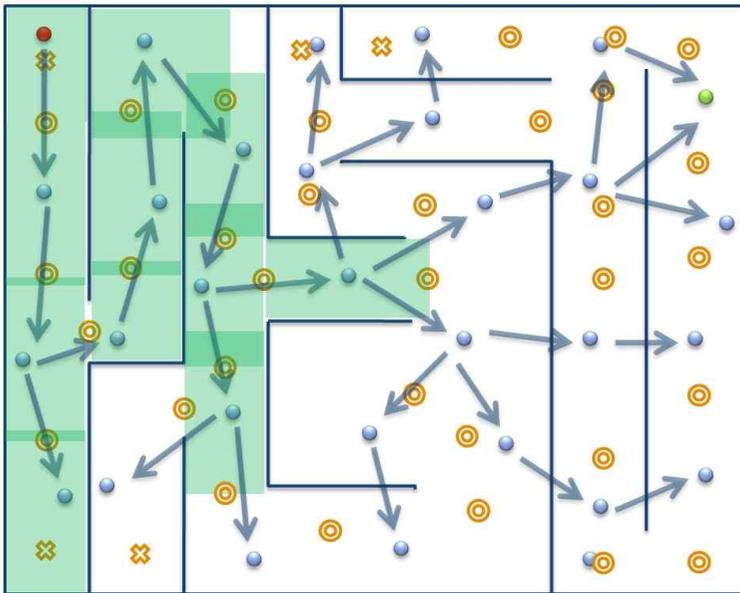
Decentralized 기법은 Source 로봇이 Search Message를 작성 전송하는 [표 1] 의 과정을 수행한다. 경로 로봇은 이 메시지를 [표 2] 과정을 수행하여 Search Message를 보내게 된다. 전 지역에 Search Message가 전송되어 목적지에 도달하게 되면 Path Message를 전송[표 3]하게 된다.

[표 1] Source 로봇의 경로탐색

1. **Search Message <- Num, Open Point, Destination Address, Relay Robots**
2. **Search Own Area**
3. **Make Open, Close Points**
4. **IF (Destination in own search area) THEN**
5. **END**
6. **ELSE**
7.     **Send Search Message to Neighbor**
8. **ENDIF**



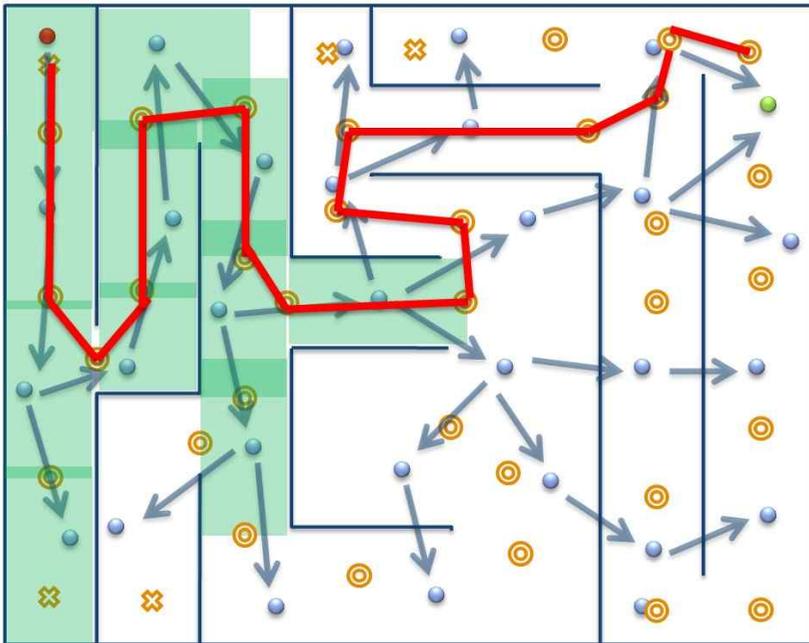
[그림 13] 보로노이 포인트 작성과 전송



[그림 14] 보로노이 포인트를 이용한 목적지 로봇까지 메시지 전송

[표 2] 경로 로봇의 Serch Message 처리과정

1. Path Message <- Relay Robots, a Whole Path
2. Search Own Area
3. Make Open, Close area
4. IF (Receive New Search Message) THEN
5.     IF (Destination in own search area) THEN
6.         Send Path Message to Source
7.     ELSE
8.         Add oneself for Search Message in Relay Robots
9.         Send Search Message to Neighbor
10.    ENDIF
11. ENDIF



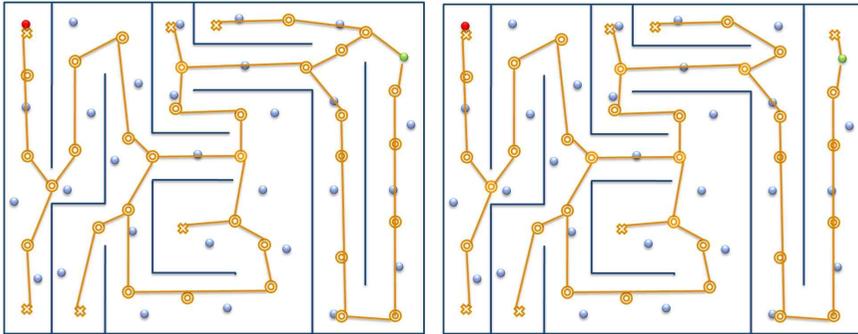
[그림 15] Decentralized 기법의 경로 설정

[표 3] 경로 로봇의 Path Message 처리과정

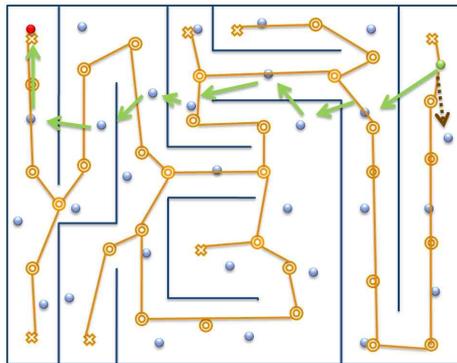
1. **WHILE ( Receive New Path Message)**
2. **IF (Last of Relay Robots in Path Message) THEN**
3.     **Wait for a little while**
4.     **Select shortest path**
5.     **Save the Path Message for a while**
6.     **Delete oneself in Relay Robots**
7.     **Add path to sender for a Whole Path**
8.     **Send Path Message**
9. **ENDIF**
10. **ENDWHILE**

### 3.3.2 장애물 변화에 따른 경로 설정

N·R의 응용은 주변 환경이 변화 할 수 있다. 로봇은 장애물이 변화에 다른 경로를 설정 할 수 있어야 한다. Centralized 방식은 경로가 두절되는 경우 Source로 이것을 알려주고 다시 경로를 설정하게 된다. 두절된 경로가 Source로부터 먼 경우 많은 로봇을 거쳐서 경로가 두절된 것을 알게 되며, 재 경로 설정을 위해 많은 메시지 전송과 계산이 필요하다. Decentralized 방식은 경로 설정에 이용한 정보를 이용하여 경로 로봇이 지역적으로 두절된 경로를 연결해 준다. 이러한 능력을 위해 경로 로봇은 두절된 경로를 발견할 경우 Repair Message를 전송[표 4]하고, 이를 받은 로봇은 자신이 알고 있는 경로를 알려준다.[표 5]



[그림 16] 장애물의 변화에 따른 경로 설정



[그림 17] Centralized Repair Message(실선) 과 Decentralized (점선)

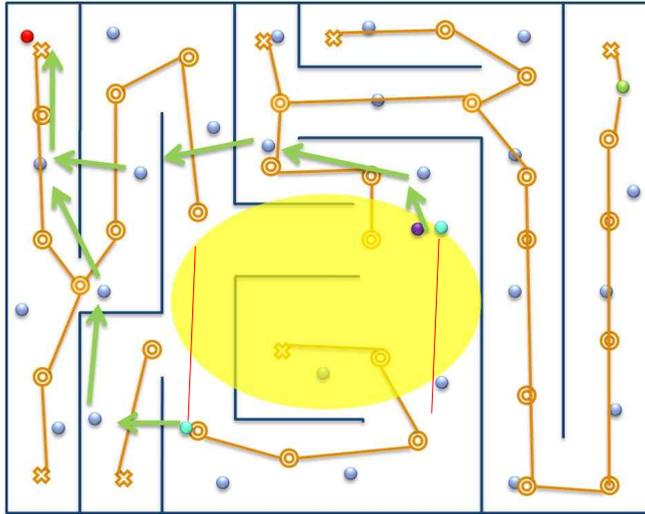
[표 4] 두절된 경로의 발견 및 메시지 전송

- |  |
|--|
| <ol style="list-style-type: none"> <li>1. WHILE (Receive New Repair Message)</li> <li>2. IF (Know about Path to Source) THEN</li> <li>3.     Send Path to Destination</li> <li>4. ELSE</li> <li>5.     Delete oneself in Relay Robots</li> <li>6.     Add path to sender for a Whole Path</li> <li>7.     Send Repair Message to Neighbor who know own open points</li> <li>8. ENDIF</li> <li>9. ENDWHILE</li> </ol> |
|--|

[표 5] 경로 로봇의 Repair Message의 처리

1. **Repair Message** <- Source, Relay Robots, a Whole Path
2. **IF (Find Broken Path) THEN**
3.     **Pull out all saved Path Message**
4.     **WHILE (Process of all Path Message)**
5.         **Delete oneself in Relay Robots**
6.         **Add path to sender for a Whole Path**
7.         **Send Repair Message to Neighbor who know own open points**
8.     **ENDWHILE**
9. **ENDIF**

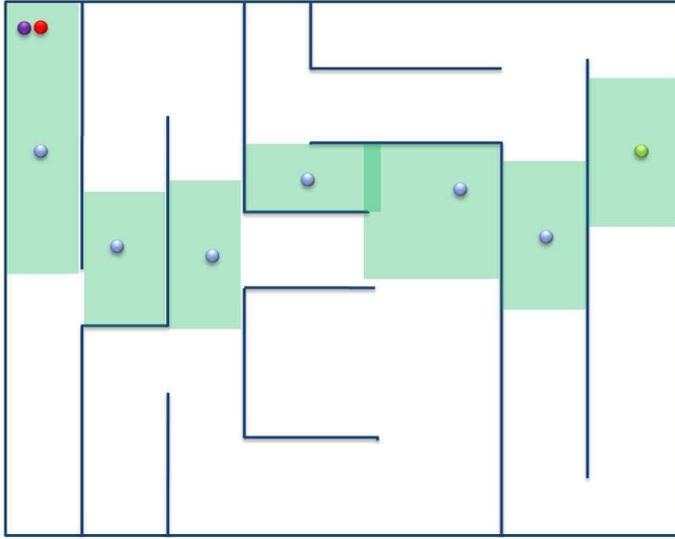
[그림 16]과 같이 경로가 변하는 경우 외에도 중간 경로가 소실되는 경우에 대한 해결책이 필요하다. Centralized 기법의 경우 탐색되지 않은 지역은 Source가 직접 이동하여 탐색하는 방법을 이용한다. 하지만 Decentralized 기법은 경로 로봇이 대신 탐색을 하여 정보를 전송해 줌에 따라[그림 18] 짧은 시간 내에 최단거리를 구할 수 있다.



[그림 18] Decentralized 기법의 경로 로봇 탐색

### 3.3.5 Initial State에서의 Centralized 기법과 Decentralized 기법 비교

적은 로봇이 넓은 공간에 있는 경우 시작 단계[그림 19]에서는 모든 로봇에게 주변 환경 정보를 받더라도 경로를 만들 수 없다. 따라서 Centralized 방식은 주어진 정보만으로 만든 전역경로를 기반으로 직접 이동하면서 경로를 만들어 나가야 한다. 하지만 Decentralized 방식은 경로 로봇이 중간 경로를 대신 찾아 줌으로서 짧은 시간과 최단경로를 가질 수 있다.

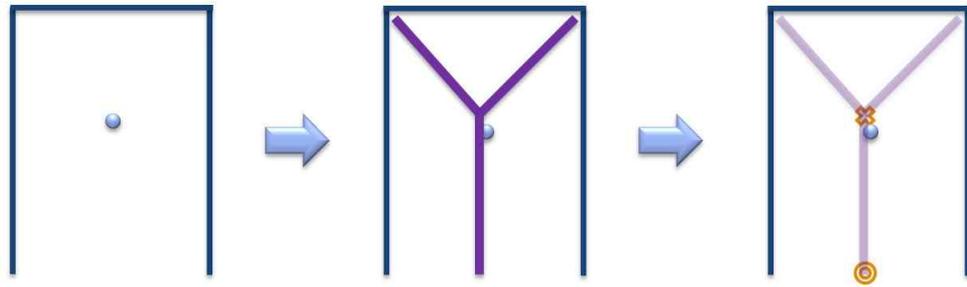


[그림 19] 시작단계에서 Centralized 기법의 주변 환경 정보

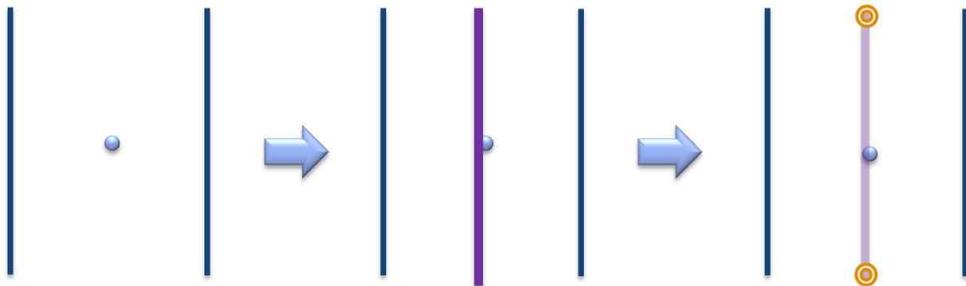
### 3.4 보로노이 포인트

분산 로봇의 경로 찾기 알고리즘에서 각 로봇이 보내는 주변 장애물 정보 메시지의 양은 매우 많다. 따라서 이 메시지의 크기를 줄임으로서 전체적인 통신량을 크게 줄일 수 있다.

본 논문에서는 보로노이 다이어그램을 이용한 보로노이 포인트를 제안한다. 기존의 보로노이 다이어그램을 이용한 경로설정은 장애물 간 중앙 경로를 설정함에 따라 충돌방지에 좋다. 이러한 장점을 이용하면서 압축된 정보를 보내기위해 각 로봇은 주변 정보를 보로노이 다이어그램을 통한 보로노이 포인트를 만들어 필요 로봇에게 전송한다.



[그림 20] 보로노이 다이어그램을 이용한 보로노이 포인트(1)



[그림 21] 보로노이 다이어그램을 이용한 보로노이 포인트(2)

로봇은 환경 정보를 보내기 위해 [그림 20, 21] 왼쪽과 같은 환경을 모두 전송한다. 이러한 방식은 많은 데이터를 전송 할 뿐 아니라, 많은 계산을 유발한다. 중앙 그림과 같은 보로노이 다이어그램을 이용하면 충돌 방지와 상대적으로 작은 데이터량을 가지지만, 오른쪽 그림과 같이 보로노이 포인트를 이용하면 더 작은 데이터로 더 작은 계산량을 가질 수 있다.

## 4. 성능평가

Decentralized 기법은 경로의 계산을 경로 로봇이 분산 계산하여 적은 시간 안에 경로를 만들 수 있다. 두절된 경로를 재 설정함에 따라 짧은 시간 내에 최단거리를 만들 수 있으며, 보로노이 포인트를 이용하여 전체 통신량을 줄인다. 4.1에서 언급된 환경에서 시뮬레이션을 통해 성능을 분석하였다.

### 4.1 시뮬레이션 환경

Centralized 기법과 제안한 Decentralized 기법을 비교 분석하였다. 공간의 크기와 로봇의 수의 변화에 따른 결과를 얻기 위해 [표 6]과 같은 4가지 시나리오에서의 성능평가를 실시하였다.

[표 6] 시뮬레이션 시나리오

Scenario 1
Few number of Robots (6 bots)
Small area (40m x 40m)
Scenario 2
Few number of Robots (6 bots)
Large area (80m x 80m)
Scenario 3
Many number of Robots (12 bots)
Small area (40m x 40m)
Scenario 4
Many number of Robots (12 bots)
Large area (80m x 80m)

정확한 성능평가 및 장애물의 변화에 따른 성능평가를 위해 각

시나리오는 [표 7]과 같이 배치가 다른 4번과 같은 배치에서 장애물이 3번 변화 한다.

[표 7] 시뮬레이션의 배치 변화와 장애물 변화

	기본	재 배치	재 배치	재 배치
기본	1	2	3	4
첫 장애물 변화	5	6	7	8
두 번째 장애물 변화	9	10	11	12
세 번째 장애물 변화	13	14	15	16

로봇이 한쪽에 배치되어 정확한 성능이 나오지 않는 경우를 막기 위해 반은 보로노이 다이어그램을 통해 배치되며, 나머지 반의 로봇은 임의로 배치된다.

로봇은 1m/s로 이동한다.

주변 정보 센싱은 실제 초음파 거리센서의 거리와 같은 3M로 가정한다.

통신거리는 실내에서의 무선랜 통신거리인 25M로 가정한다.

모든 로봇은 절대 위치를 오차 없이 알고 있으며, 장애물 센싱에 오차가 없다고 가정한다.

시뮬레이션은 C++을 이용하여 구현하였다.

Decentralized 기법은 Open Point로 이루어진 경로[식 1]를 이용하기 때문에 [식 2]를 이용하여 각 로봇의 이동거리를 계산한다. 모든 로봇의 이동거리는 [식 3]을 이용한다.

[식 1] Path of source (Decentralized)

$$O_{i,j}^{(1)} \dots O_{i,j}^{(k)}$$

[식 2] A length of path (Decentralized)

$$P_i = \sum_{i=1}^{N-1} \| O_{i+1,j+1}^{(k)} - O_{i,j}^{(k)} \|$$

[식 3] Distance of all Robots (Decentralized)

$$T = \sum_{i=1}^N P_i$$

Centralized 기법은 주변 환경정보를 그대로 전송하기 때문에 Decentralized 기법보다 많은 전송량을 가지게 된다. 전송량은 같은 크기의 Request와 다른 크기의 Ack의 합으로 구해진다.[식4, 5]

[식 4] Centralized ack traffic

$$CA = \sum_{i=1}^N C_i$$

[식 5] Decentralized ack traffic

$$DA = \sum_{i=1}^N D_i$$

$C = \text{Centralized Ack}$

$D = \text{Decentralized Ack}$

[식 6] Sum of Centralized traffic

$$TC = \alpha \cdot \sum R_i + \beta \cdot \sum CA_i$$

[식 7] Sum of Decentralized traffic

$$TD = \alpha \cdot \sum R_i + \gamma \cdot \sum CD_i$$

$\alpha = \text{Size of Request Message}$

$\beta = \text{Size of Centralized Ack Message}$

$\gamma = \text{Size of Decentralized Ack Message}$

계산량은 각 Ack에 대한 데이터 크기를 곱하여[식8, 9] 얻을 수 있다.

[식 8] Sum of centralized computation

$$CC = \lambda \cdot \sum C_i$$

[식 9] Sum of decentralized computation

$$CD = \mu \cdot \sum D_i$$

$\lambda = \text{Requirement for CA}$

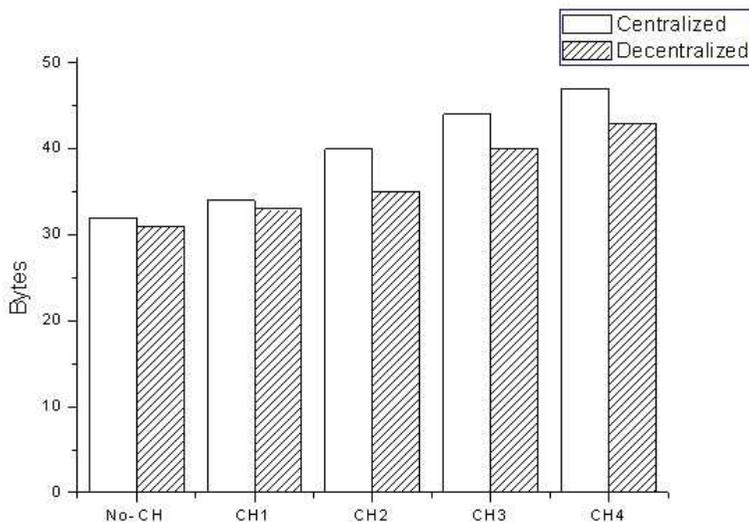
$\mu = \text{Requirement for DA}$

## 4.2 결과 및 분석

4.1에서 언급된 환경에서 시뮬레이션 모델과 시나리오를 통해 성능을 분석한 결과 로봇이 많을수록, 장애물 지도가 커질수록 Decentralized 기법이 Centralized 기법보다 좋은 성능을 보였다.

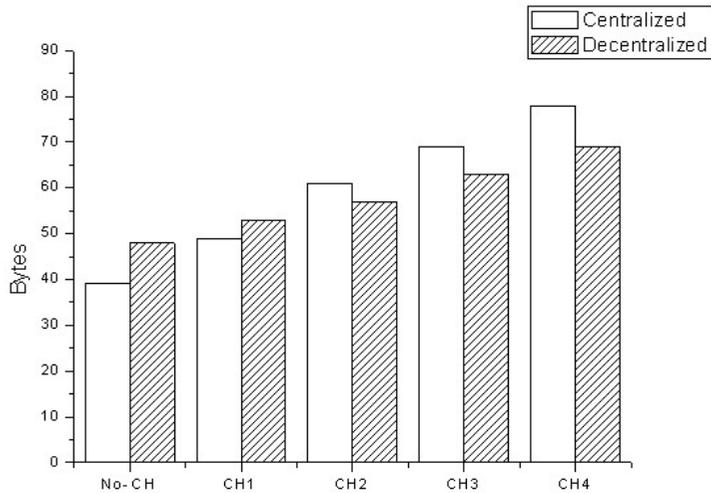
### 4.2.1 통신량

제안된 경로 찾기 알고리즘은 보로노이 포인트를 이용하여 같은 크기의 환경 정보를 전송할 때 데이터의 크기를 줄였다. 그래프 상에서는 상대적으로 적은 통신량을 보이지만 Centralized 기법은 많은 공간을 고려하지 않기 때문에 가장 중요한 최단 경로를 구하지 못하였다. 하지만 Decentralized 기법은 탐색되지 않은 지역까지 정보를 전송하여 통신량은 늘어났지만 최단경로를 구할 수 있었다.



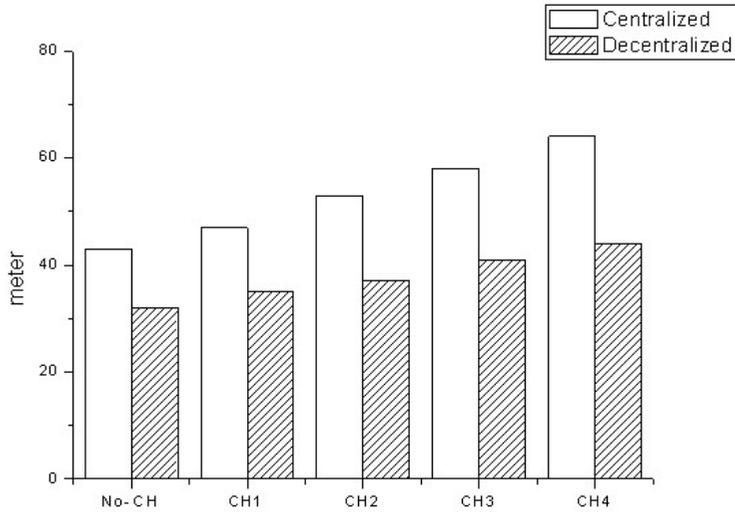
[그림 22] Scenario 1의 총 통신량

적은 수의 로봇과 작은 공간으로 이루어진 시나리오1 에서의 통신량[그림 22]는 Centralized기법에서 고려된 환경의 크기가 Decentralized기법에서 고려된 환경 크기에 비해 작고, 같은 환경정보를 보낼 때 Decentralized기법이 적은 데이터를 보내기 때문에 큰 차이는 보이지 않는다.



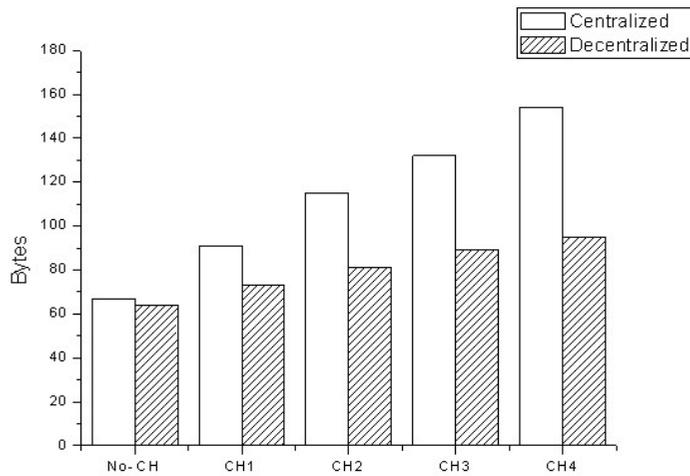
[그림 23] Scenario 2의 총 통신량

넓은 환경에 적은수의 로봇의 이동[그림 23]에서는 변화가 없을 때에는 Decentralized기법이 많은 환경을 고려함으로 더 많은 데이터 전송을 보이지만, Centralized기법은 적은 환경을 고려함에도 불구하고 변화할 때마다 더 많은 데이터 전송을 요구함을 알 수 있다.



[그림 24] Scenario 3의 총 통신량

[그림 22]와 같은 수의 로봇이 전체 환경 크기가 커진[그림 24] 환경에서는 Centralized기법이 고려된 환경의 크기가 작음에도 불구하고 많은 이동으로 인해 많은 데이터 전송량을 보인다.

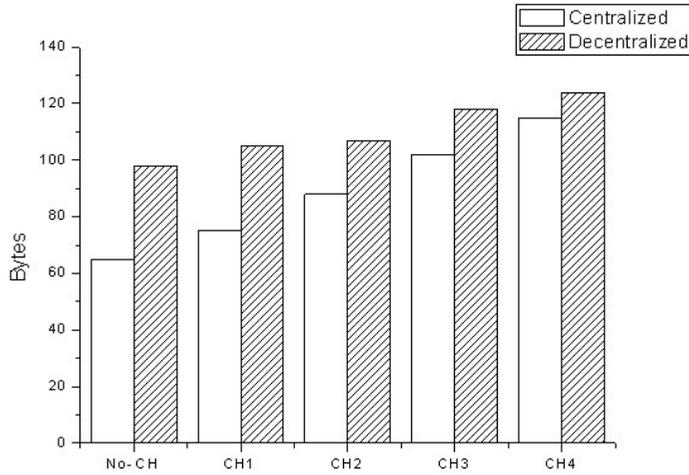


[그림 25] Scenario 4의 총 통신량

넓은 공간에 많은 로봇이 있는 환경에서[그림 25]는 Decentralized 기법이 줄어든 환경 데이터를 이용함으로써 많은 공간을 고려했음에도 불구하고 변화하지 않을 때도 Centralized 기법과 비슷한 전송량을 보였다. 그리고 전체 환경이 변화함에 따라 지역적인 해결을 이용하여 큰 통신량의 증가 없이 경로를 찾을 수 있었다.

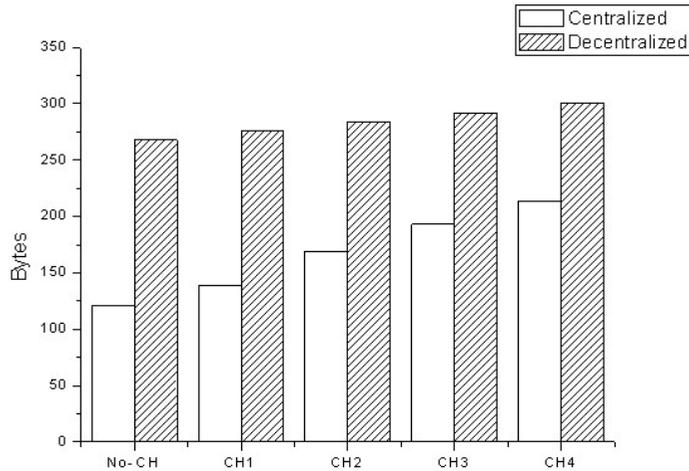
#### 4.2.2 계산량

제안된 경로 찾기 알고리즘은 보로노이 포인트를 이용하여 같은 크기의 환경 정보를 전송할 때 데이터의 크기를 줄였다. 이를 위해서 전송하기 전에 보로노이 포인트를 구하는 오버헤드를 가진다. 하지만 제안된 알고리즘은 A\* 알고리즘에 비해 복잡도가 낮다. Centralized 기법은 탐색되지 않은 공간을 고려하지 않기 때문에 그래프 상에서 볼 때 작은 공간에서는 Centralized 기법과 비슷한 계산량을 가진다. 그리고 넓은 공간에서는 계산량이 많이 줄어든 것을 볼 수 있다. 계산량의 결과를 분석하면 보로노이 포인트를 이용하였음에도 불구하고 더 많은 계산량을 보인다. 하지만 더 많은 지역을 계산함에 따라 생기는 계산량이기 때문에 더 중요한 요소인 최단거리를 구하기 위한 Trade-Off 이다.



[그림 26] Scenario 1의 총 계산량

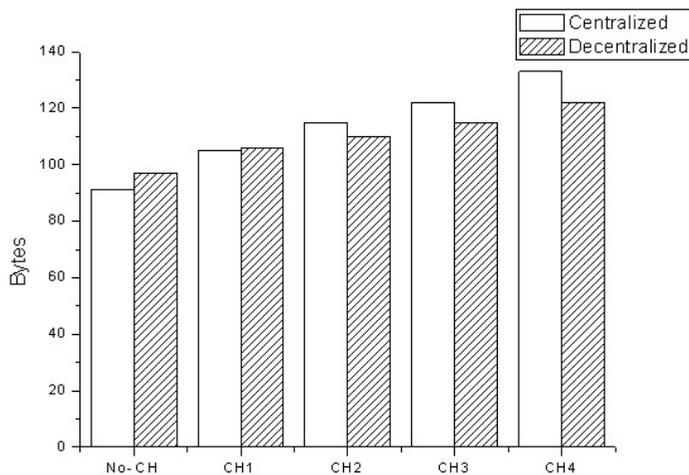
많은 주변정보를 처리함에 따라 Decentralized기법이 더 많은 계산량을 가진다. 하지만 한 곳에서 계산되는 Centralized기법과 달리 경로에 있는 로봇들의 계산량합이기 때문에 실제 한 로봇의 계산량은 크게 줄어들게 되며, 이를 통해 짧은 시간 내에 결과를 구할 수 있다.



[그림 27] Scenario 2의 총 계산량

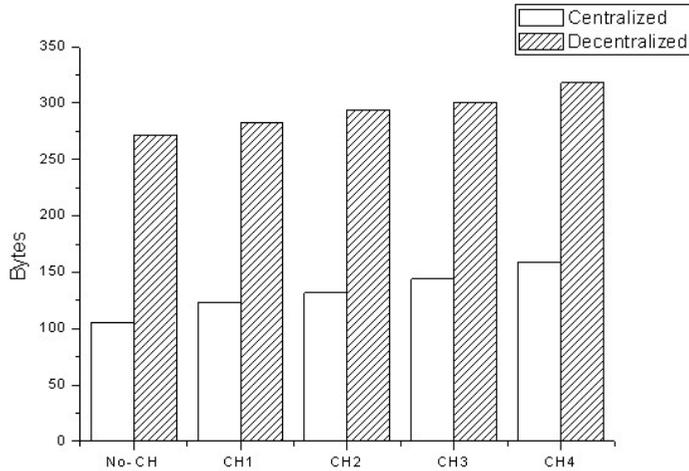
Centralized기법의 넓은 공간에서의 이동은 두절된 공간이동시 통신을 이용하지 않기 때문에 통신량이 줄어들며, 또한 적은 환경내의 경로를 설정하기 때문에 적은 계산량을 가진다.

Decentralized기법[그림 27]은 넓은 공간을 고려하여 최단거리를 구하기 위해 많은 계산량을 가지지만, 분산된 계산을 통하여 짧은 계산시간을 가질 수 있다.



[그림 28] Scenario 3의 총 계산량

작은 공간에 많은 로봇에 의해 거의 모든 공간이 고려되는 경우 [그림 28]에는 비슷한 계산량을 보인다. 변화가 없는 경우에는 Decentralized기법이 약간 더 많은 계산을 하지만 변화함에 따라 Centralized기법보다 적은 계산량을 가진다.

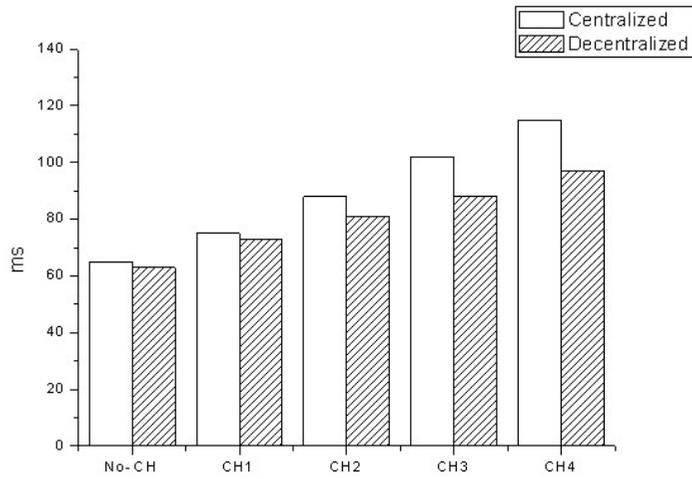


[그림 29] Scenario 4의 총 계산량

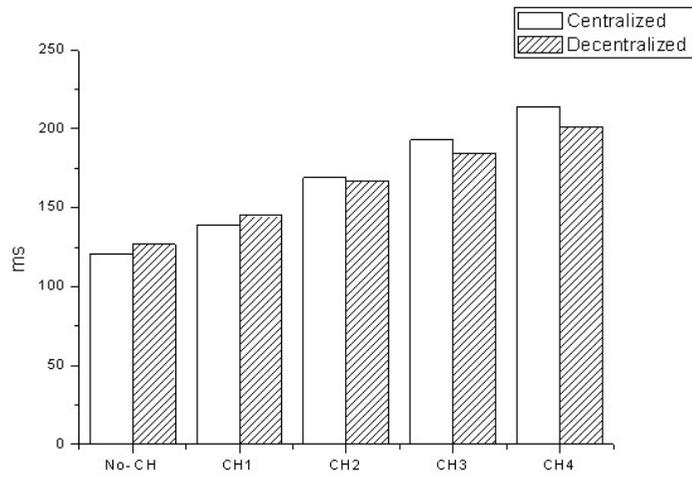
[그림29]에서 넓은 공간에서는 적은 환경만을 고려되기 때문에 Centralized기법이 적은 계산량을 가지지만, 많은 환경을 고려하여 로봇의 이동에서 중요한 최단거리를 구하기 위한 오버헤드이다.

### 4.2.3 계산 시간

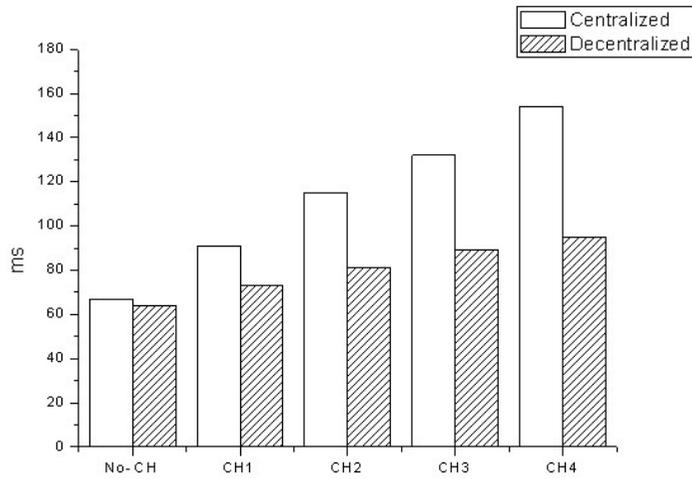
로봇의 경로 찾기 알고리즘에서 제한된 시간내에 결과를 구하는 것은 중요한 요소이다. 제안된 알고리즘은 한 로봇에서의 집중된 계산이 아닌 분산된 계산을 통해 빠른 결과를 얻을 수 있으며 결과 분석을 통해 있음을 알 수 있다. [그림 30, 31, 32, 33]에서 Centralized기법과 Decentralized기법은 비슷한 계산시간을 보인다. Centralized기법은 한 곳에서 계산된 시간이지만 Decentralized기법은 경로 로봇의 계산 시간의 총합이기 때문에 분산 계산을 통해 짧은 시간내에 결과를 얻을 수 있다.



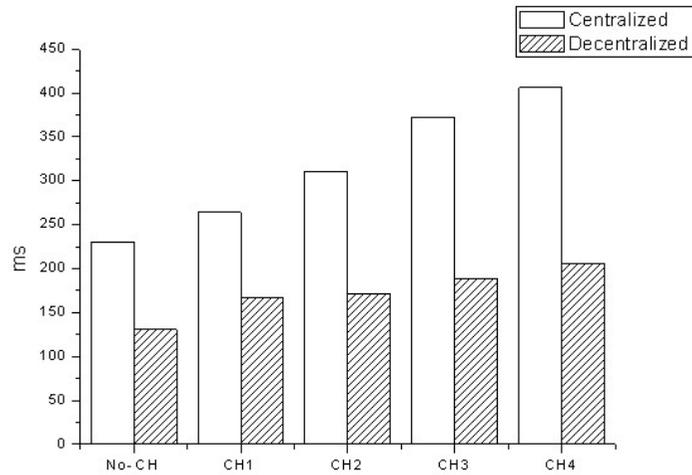
[그림 30] Scenario 1의 총 계산 시간



[그림 31] Scenario 2의 총 계산 시간



[그림 32] Scenario 3의 총 계산 시간

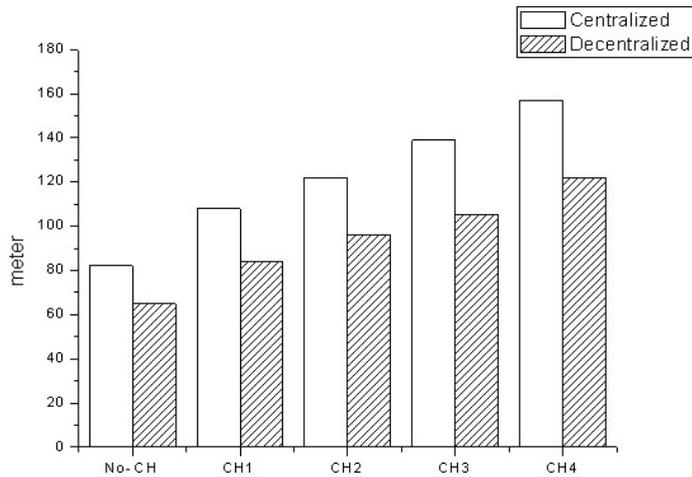


[그림 33] Scenario 4의 총 계산 시간

많은 수의 로봇에서는 고려된 공간[그림 32, 33]이 많아 짐에 따라 Centralized 기법이 많은 시간을 보이는 것에 비해 Decentralized 기법이 적은 계산시간을 가짐을 확인할 수 있다.

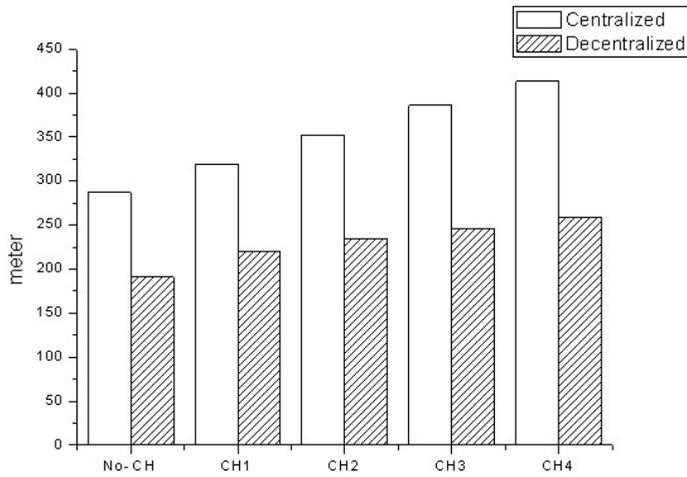
## 4.2.4 이동 거리

로봇 계산능력과 통신능력, 에너지의 제약이 적은 편이기 때문에 경로 찾기에서 최단거리를 찾는 것은 가장 중요한 요소이다. 제안하는 알고리즘은 짧은 시간내에 많은 공간에 대한 계산을 통하여 최단거리를 구한다. [그림34, 35, 36 ,37]을 통해 작은 공간에 많은 로봇이 있는 경우에는 큰 차이가 없지만, 로봇이 적을 때도 넓은 공간에서 최단거리를 구할 수 있다.



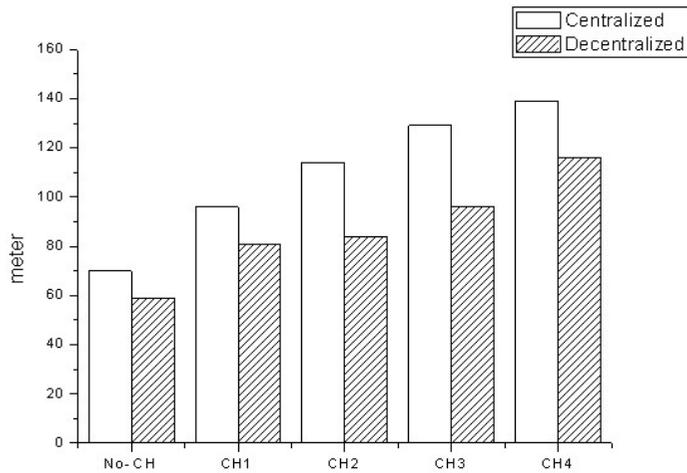
[그림 34] Scenario 1의 이동 거리

[그림 34]를 통해 넓은 환경을 고려하여 변화가 없을 때 짧은 경로를 얻을 수 있으며, 변화하는 환경에서도 큰 거리의 증가가 없음을 알 수 있다.



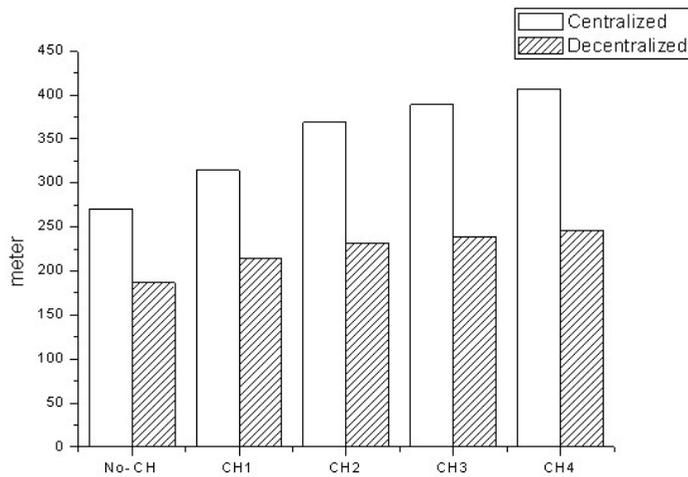
[그림 35] Scenario 2의 이동 거리

[그림 35]를 통해 연결된 경로를 찾지 못함으로 인해 환경정보를 받지 못하고 이동함으로서 Centralized기법은 큰 이동거리를 가진다.



[그림 36] Scenario 3의 이동 거리

[그림 36]은 [그림 34]에 비하여 이미 모든 환경을 고려한 Decentralized기법은 이동거리가 짧아지지 않았으나, 많은 수의 로봇으로 Centralized기법의 이동거리가 줄어들었음을 알 수 있다.



[그림 37] Scenario 4의 이동 거리

Centralized기법은 [그림 35]에 비해서 [그림37]에서 로봇의 증가에 따라 약간의 거리가 줄어들었지만 로봇이 두 배 증가함에도 불구하고 큰 이동거리의 감소를 볼 수 없다. 하지만, Decentralized기법은 많은 환경을 고려함에 따라 환경 변화가 없을때도 짧은 경로를 구할 수 있었으며, 환경이 변화함에도 큰 거리의 변화 없이 목적지까지 경로를 구할 수 있다.

## 5. 결론

본 논문은 변화하는 환경에서 로봇의 협력, 협동을 통하여 효과적인 일 처리를 하는 Networked Robots 시스템을 위한 분산 로봇 경로 찾기 알고리즘을 제안 한다.

많은 로봇과 장치로 이루어진 분산 환경에서 발생한 정보의 전송, 처리 및 명령을 내리는 Networked Robots 시스템을 제안하고, 적은 통신량과 계산량으로 최적경로를 설정한다.

제안된 Decentralized 방식은 경로의 설정 단계에서 분산된 계산을 통해 넓은 지역과 많은 로봇에 확장성을 가지고 있다. 또한 두 절된 경로와 변화하는 장애물 지도에서도 최적해를 구할 수 있다. Centralized기법의 문제인 확장성과 제안된 시간내에 해를 구할 수 있으며 시뮬레이션 성능 분석결과 확장성, 적은 통신량, 적은 계산량을 가지는 좋은 성능을 보임을 알 수 있다.

## 6. 참고문헌

- [1] The IEEE Robotics and Automation Society:  
<http://www.ieee-ras.org>. Accessibility verified Dec 19, 2009.
- [2] V.Kumar, D. Rus, and G. S. Sukhatme, “Networked robots,” in Handbook of Robotics, B. Siciliano and O. Khatib, Eds. New York: Springer-Verlag, 2008, ch. 41, pp. 943 - 958.
- [3] O. Tekdas and V. Isler, “Robotic Routers,” Proc. IEEE Int’l. Conf. Robotics Automation, May 2008, pp. 1513 - 18.
- [4] Kiva Systems:  
<http://www.kivasystems.com>. Accessibility verified Dec 19, 2009.
- [5] Maxim A. Batalin, Gaurav S. Sukhatme and Myron Hattig, “Mobile Robot Navigation using a Sensor Network,” in IEEE Int. Conf. on Robotics and Automation, 2003.
- [6] A. Batalin and S. Sukhatme, “Mobile robot navigation using a sensor network,” IEEE International Conference on Robotics and Automation, pp. 636-642, 2003.
- [7] Wang Mei, Wu Tiejun, “Cooperative Co-evolution Based Distributed Path Planning of Multiple Mobile Robots”, Journal of Zhejiang University SCIENCE, 2005 6A (7), pp.697-706.

[8] M. Bennewitz, W. Burgard, and S. Thrun. Optimizing schedules for prioritized path planning of multi-robot systems. In Int. Conf. on Robotics and Automation (ICRA'01), Seoul, Korea, 2001.

[9] F. Aurenhammer, "Voronoi Diagrams", ACM Computing Surveys 23 (1991)345-405.

[10] S. Garrido, L. Moreno, M. Abderrahim, F. Martin, Path planning for mobile robot navigation using voronoi diagram and fast marching., in: Proc of IROS'06, Beijing, China, 2006, pp. 2376 - 2381.

[11] Yang Dongyong, Chen Jinyin, Matsumoto, N., Yamane, Y., "Multi-robot Path Planning Based on Cooperative Co-evolution and Adaptive CGA, Intelligent Agent Technology, 2006, pp.547-550.

[12] Network Robot Technology:  
[http://www.etri.re.kr/eng/etri/sub02/sub02\\_4102\\_1.jsp](http://www.etri.re.kr/eng/etri/sub02/sub02_4102_1.jsp). Accessibility verified Dec 19, 2009.

[13] Tekdas, O., Lim, J., Terzis, A., Isler, V.: Using mobile robots to harvest data from sensor fields. IEEE Wireless Communications (2009)

[14] W. Yang. An algorithm for network formation and an implementation of a mobile robotic router system. Master's thesis, Rensselaer Polytechnic Institute, 2008.

## Abstract

# A Decentralized Robot Navigation Algorithm in Dynamic Environments

Hak Soo KIM

School of Electronics and Information

Kyung Hee University

**Abstract.** In recent years, robotics has developed over a cooperation and coordination with many robots. Robots needs communication skills and the ability to move the location for Cooperation and collaboration.

In this paper presents a networked robots system and decentralized path planning algorithm for multiple robots in large areas. The existing methods for path planning for multiple robots has two solution. In the centralized scheme can find best way to destination. but, Have two problems. First, Inflexible and Not suitable for environments changes or uncertainties. Because one robot computes a path to destination, Their complexity is high and exponential increase in large space. Communication is similar to that of the scalability. Second, Increase communicate amounts for compute a path. Decentralized methods using a localization message. Therefore, Decrease communication amount.

In this paper we consider the problem of finding a best way

for multiple robots in dynamic environments. We propose decentralized path planning algorithm. Decentralized methods is able to decrease communication amounts and low complexity.

Keyword : Robot Navigation, Networked Robots, Distributed Navigation, Voronoi Diagram

## 감사의 글

짧다면 짧은, 길다고 생각하면 매우 긴 2년이 지나 석사과정이 마치게 되었습니다. 혼자 정장을 입지 않고 면접을 보면서 시작한지가 어제 같은데 벌써 졸업을 눈앞에 다가오니 많은 생각을 하게 됩니다. 먼저 석사과정까지 마칠 수 있는 환경을 만들어주신, 낱아주시고 길러주신, 사랑을 주시는 부모님께 감사드립니다. 그리고 부족한 저에게 많은 것을 가르쳐 주시며 지도해주신, 학문 뿐 아니라 인생에 필요한 마음가짐과 자세를 알려주신 조진성 교수님께 깊은 감사드립니다. 바쁘신 가운데 논문 심사를 해 주신 홍충선 교수님, 이성원 교수님 그리고, 수업을 가르쳐주신 교수님들에게 감사 드립니다.

석사과정동안 대부분의 시간을 같이 보낸 연구실의 같은 모바일 팀이자 기독교인으로써 모범이 되는, 같이 수영도 다니고 많은 애기를 해준 충용형, 연구실 연구 및 과제에서 많은 도움을 주신, 그리고 랩장으로서 많은 도움을 주신 대영형, 침에 들어와 많은 공부를 가르쳐준 권택형, 항상 다툼이 많았지만 정말 사람 좋다고 모두가 인정한 의연형, 연구실에서 열정이 넘치고 언제나 개그본능에 충실한 요한형, 처음부터 반갑게 맞아주고, 열심히 하는 모습을 좋아하는, 농구를 사랑하는 경원, 뒷자리에서 많은 도움을 준 귀로, 동기이자 많은 일과 정말 많은 연구를 한, 그리고 2년간의 강의의 열매를 얻은 영선, 연구실에 처음 들어와 옆자리에서 많은 도움을 준 용규, 유퍹케어를 같이 진행한, 웃음과 열정이 가득한 준성, 이제 연구실을 이끌어 나가게 될 아끼는 후배들인 일년동안 막내하느라 고생이 많은, 그리고 많은 시간을 같이한 범석, 연구실에서 잊고 있던 많이 익숙한 느낌을 가진, 센스있고 열심히 상배, 마지막 학기에 많은 도움을 주지 못해 미안함이 남는, 정말 열심히 공부하는

내년의 품절남 자 룡, 연구실 생활을 시작하는 재우에게 감사드립니다. 그리고, 제가 들어오기 전에 좋은 연구실을 만들어 주신 졸업한 두경형, 건백형, 준하형, 경환형, 형관형, 천환형, 헌준형, 상하, 재호, 승민, 정현 모두에게 감사를 드립니다. 이미 졸업해서 특별한 날에만 본 형들도 많지만 연구실에 필요한 많은 연구와 금전적인 여유를 남겨 주셔서 대단히 감사합니다. 연구실 생활 외에도 멀관 칼럼에서 근무와 임베디드, 유닉스, 소프트웨어공학 수업조교를 하면서 알게된 모든 분들에게 감사드립니다. 항상 지켜주시며, 함께해 주시는 하나님 아버지께 감사드립니다. 연구실 생활동안 혼자 바쁜척은 다하며 연락도 잘 안한 친구였는데 먼저 연락을 주며 일정에 맞춰 만나며 좋은 우정 가진 정호, 동현, 태승 그리고 진욱, 동훈, 종길, 민영, 찬수, 용덕, 형준, 상철, 진욱, 성용, 근태, 준형, 현상과 새로운 우정을 만든 광현, 한근에게 감사 드립니다. 짧은 글로 석사과정동안 생활할 수 있는 많은 것을 적으려고 하니 너무나도 부족함이 느껴집니다. 이 짧은글로 저의 감사함을 다 표연할 수 없지만, 아버지, 어머니, 할아버지, 할머니, 그리고 누나에게 깊은 감사드리며 이 논문을 바칩니다.

2010년 겨울 김 학 수 드림