

모바일 로봇을 위한 다단계 형태 공간 생성 방법에 대한 성능 검증

Performance Evaluation for Multi-Level Configuration Space for the Development of an Efficient Mobile Robot Path Planner

박정수 · 소병철 · 정진우*

Jung-Soo Park, Byung-Chul So and Jin-Woo Jung

동국대학교 컴퓨터공학과

요 약

모바일 로봇의 경로 계획을 위해 형태 공간(Configuration space)과 형태 장애물(Configuration obstacle)이라는 개념이 많이 활용되고 있다. 이 개념은 이동로봇을 공간 상에서 하나의 점으로 간주할 수 있도록 주변 장애물을 확장시킨다는 것으로, 이를 통해 장애물과의 충돌로부터 자유로운(Collision free) 이동 경로를 쉽게 찾아낼 수 있게 된다. 또한, 이러한 형태 공간 및 형태 장애물을 쉽게 생성하는 가장 보편적인 방법 중 하나는 이동 로봇의 형태를 원형으로 근사화하는 것이다. 이는 그 방법이 간단하기 때문에 이동 로봇의 구체적인 형태 및 이동 메커니즘을 고려하여 형태 공간을 생성하는 방법보다 형태장애물 생성 시간을 크게 단축시킬 수 있게 해준다. 하지만 이동 로봇을 원형으로 근사화하여 형태 장애물을 생성할 경우 비교적 가까이에 있는 실제 장애물들이 하나의 형태 장애물로 병합될 수 있다는 문제점이 있다. 이로 인하여 형태 공간내에서 이동 경로를 생성할 경우 최적의 이동 경로를 찾는다는 보장을 할 수가 없게 된다. 따라서 형태 공간 내에서 최적에 가까운 이동 경로를 효율적으로 찾으려 하기 위해 부분적으로 보다 정확한 형태 공간을 생성하면서도 전체 생성 시간을 단축시킬 수 있는 다단계 형태 공간 생성 방법을 제안하였다. 제안된 방법은 로봇을 원형으로 근사화시킨 뒤 시작 지점과 목표 지점을 잇는 이상적인 경로를 생성하고 이 경로 상에 존재하는 형태 장애물이 로봇의 원형근사화로 인해 주변의 다른 형태 장애물과 병합되었다면 해당 형태 장애물에 대해서만 보다 정확한 형태 장애물을 재 생성한다는 방법이다. 또한, 본 논문에서는 기존의 정확한 형태 공간 생성 방법과 새롭게 제안한 다단계 근사화 형태 공간 생성 방법을 비교하기 위해 다양한 이동 로봇의 형태와 회전 각도에 대해 형태 공간을 생성하는데 소요되는 생성 시간을 비교·분석해 보았다.

키워드 : 형태 공간, 형태 장애물, 모바일 로봇, 성능 검증

Abstract

Configuration space(C-space) including configuration obstacle(C-obstacle) is one of the most important concepts in mobile robot path planning. Using C-space and C-obstacles, the robot with different shapes and moving mechanisms can be considered as a point in the C-space. And, as a result, the collision-free path for the robot can be easily achieved. To make C-space including C-obstacle, many researchers used circular approximation method for the efficient calculation time. This method can help us to save our time by approximating the shape of a robot as the minimum sized circle which can cover all the area of robot. But, by using the circle larger than the robot, more space are considered as the part of robot and, as a result, some obstacles which are very near each other may be considered as a combined one obstacle. To solve this problem, multi-level configuration space is proposed by this paper. This multi-level method also use the circular approximation method as the initial step. But, after finding the initial path, it will check how many obstacles are combined. And then, for each combined obstacle, more accurate C-space generation will be continued. To check the efficiency of the proposed algorithm, time for c-space generation are compared with the well-known accurate C-space generation method using various types of robot shape.

Key Words : configuration space, configuration obstacle, mobile robot, performance evaluation

접수일자 : 2010년 10월 21일

완료일자 : 2010년 12월 24일

*본 논문은 본 학회 2010년도 추계학술대회에서 선정된 우수논문입니다.

*본 연구는 동국대학교와 산학협력중심대학 육성사업 기술개발과제(세과과제명: RoboticsLab 시뮬레이터의 기능개선을 위한 이동로봇의 효율적 경로계획알고리즘 개발)의 지원으로 수행되었음.

+ 교신저자

1. 서 론

최근 로봇 관련 기술들의 연구 성과가 확산되면서 로봇 기술에 대한 활용영역이 산업 분야뿐만 아니라 의료용, 교육용, 가정용, 심해용, 우주용 등의 다양한 분야로 확대되고 있다.[1]. 이렇게 로봇 기술의 활용 범위가 넓어짐에 따라 이동 로봇의 안전한 주행(Navigation)이 날로 강조되고 있다[5]. 특히, 최근에는 동적으로 움직이는 장애물이 존재하거나 목표 지점이 변화하는 동적인 환경에서의 이동 로봇의

성공적인 주행을 보장하는 제어 알고리즘에 관한 연구들이 활발히 진행되고 있다[5][6][7][8][9]. 이러한 이동 로봇의 안전한 주행을 목적으로 하는 기본적인 연구 중의 하나는 장애물과의 충돌로부터 자유로우면서(Collision free) 시작 지점과 목표 지점까지의 연결된 경로를 생성해주는 경로 계획(Path planning)이다[5].

또한, 이동 로봇의 경로 계획을 수행하기 위한 중요한 개념들 중 하나가 형태 공간(Configuration space, C-space)과 형태장애물(Configuration obstacle, C-obstacle)이다. 참고로 형태 공간상에서 가상으로 그 크기가 확장된 장애물들을 형태 장애물(Configuration obstacle, C-obstacle)이라고 부른다.

실제로 이동 로봇들은 각각 다른 형태를 가질 수 있으므로 결과적으로 시작 지점에서 목표 지점으로 향하는 이동 로봇의 경로가 그 이동 로봇의 형태 및 이동 메커니즘에 따라서 달라질 수 있다는 문제를 야기한다. 이때, 로봇의 원형 근사화를 고려한 형태장애물과 형태 공간은 이러한 문제점에 대한 하나의 해결책이 될 수 있다.

구체적으로 형태 장애물을 포함한 형태 공간은 이동 로봇의 형태 및 이동 메커니즘에 기반하여 공간상에 존재하는 장애물들을 확장해 생성되었으므로, 이동 로봇을 그 형태에 상관없이 하나의 점으로 인지할 수 있게 해준다. 이 때 형태 공간의 차원의 수는 이동 로봇이 가지는 자유도(Degree of freedom)의 개수와 같다.

이러한 형태 공간과 형태 장애물이라는 개념은 이동 로봇의 경로 계획에 있어서 장애물과의 충돌에서 자유로운(Collision free) 이동 경로를 효율적으로 찾아낼 수 있는 기반이 된다. 하지만 정확한 형태 장애물을 포함한 형태 공간의 생성에는 공간상에 존재하는 모든 장애물에 대하여 로봇의 형태 및 이동 메커니즘을 고려할 필요가 있으므로 그 계산 시간이 오래 걸리게 되는 부담이 있다. 이를 해결할 수 있는 한 방법으로는 이동 로봇의 형태를 이동 로봇 전체를 포함할 수 있는 최소 크기의 원으로 근사화시키는 방법이 있다. 이는 형태 공간 생성 시 이동 로봇의 형태 및 이동 메커니즘을 고려할 필요가 없어지게 되므로 보다 쉽고 빠르게 형태 공간을 생성할 수 있게 된다.

하지만 이동 로봇을 원형으로 근사화한 형태는 기존의 이동 로봇 형태보다 더 많은 공간을 차지하게 된다. 즉, 이동 로봇의 원형 근사화를 통해 생성된 형태 공간 내부에 있는 형태 장애물은 이동 로봇의 형태 및 이동 메커니즘을 고려하여 생성한 형태 공간내의 형태 장애물보다 그 크기가 증가하게 된다는 것이다. 이러한 형태 장애물 크기의 증가는 근접해 있는 장애물들이 하나의 형태 장애물로 병합되는 문제를 발생시킬 수 있으며, 이는 이동 로봇의 경로 계획에 있어서 시작 지점과 목표 지점을 연결하는 최적의 이동 경로를 찾는 데 방해가 될 수 있다. 따라서 본 논문에서는 이러한 문제점을 해결하기 위하여 이동 로봇의 경로 계획을 위한 형태 공간 생성 시 다단계로 형태 공간을 생성하는 방법을 제안하였다[4].

2. 다단계 형태 공간 생성 방법

형태 공간을 다단계로 생성하는 방법은 크게 두 가지 단계로 나눌 수 있다. 첫 번째 단계에서는 공간상에 존재하는 이동 로봇의 형태를 원형으로 근사화하는 방법을 사용하여 이동 로봇이 이동 경로를 계획할 수 있도록 해준다. 이는

기존의 형태 공간 생성 시 이동 로봇의 형태 및 모든 자유도를 고려하는 방법보다 형태 공간을 생성하는데 필요로 하는 계산량을 줄이기 위함이다.

두 번째 단계에서는 이동 로봇의 실제 형태 및 모든 자유도를 고려한 형태 장애물의 생성을 형태 공간상에서 부분적으로 적용시킴으로써 첫 번째 단계에서 생성한 형태 공간에서 보다 최적에 가까운 이동 경로를 찾을 수 있도록 한다. 구체적인 알고리즘은 다음과 같다.

[다단계 형태 공간 생성 방법 알고리즘]

- ① 현재 환경 상에 존재하는 장애물의 개수(N1)를 파악한다.
- ② 이동 로봇의 형태를 원형으로 가정하고 1단계 형태 장애물(CO1) 및 1단계 형태 공간(CS1)을 생성한다.
- ③ ②에서 생성된 1단계 형태 장애물(CO1)의 개수(N2)를 파악한다. 또한, 2개 이상의 장애물들이 서로 병합된 경우 이를 별도로 표시해둔다.
- ④ $N1=N2$ 인 경우
 - A. CO1 및 CS1을 이용하여 경로 계획을 수행한다.
- ⑤ $N1 \neq N2$ 인 경우
 - A. 시작점(SP)과 목표점(GP)을 잇는 이상적 경로인 직선(IP)을 긋는다.
 - B. 두 개 이상의 장애물이 병합된 형태 장애물($co1 \in CO1$)에 대하여
 - i. 만약 $co1$ 과 IP가 서로 교차한다면
 - 1) $co1$ 을 구성하는 장애물들에 한정하여 2단계 형태 공간(CS2)을 생성한다. 이는 이동 로봇의 실제 형태와 이동 메커니즘을 고려한 형태 장애물(CO2)을 가지게 된다.
 - 2) CS1을 1)에서 생성된 CO2를 이용하여 새로운 CS1'을 생성한다. 이를 수학적 식으로 표현하면 $CS1' = (CS1 - \{co1\}) \cup CO2$ 이 되며, 이는 ②에서 생성된 1단계 형태 공간에서 $co1$ 의 집합을 제외한 다음 1)에서 생성된 CO2를 합하여 새로운 CS1' 형태 공간을 생성하는 것을 의미한다.
 - 3) 새롭게 생성된 CS1' 형태 공간에 대해서 이동 경로 계획을 수행한다.
 - ii. $co1$ 과 IP가 서로 교차하지 않는다면
 - 1) CO1 및 CS1을 이용하여 경로 계획을 수행한다.

그림 1은 이동 로봇의 원형 근사화를 통하여 형태 공간을 생성한 뒤 이동 경로 계획을 수행한 모습이다. 이동 로봇의 원형 근사화를 통하여 형태 장애물을 생성하면서 실제 장애물 1번과 2번, 실제 장애물 4번과 5번이 하나의 형태 장애물로 생성된 것을 확인할 수 있을 것이다. 이 형태 공간을 이용하여 이동 경로 계획을 수행하였을 경우 그림 1과 같이 형태 장애물을 우회하게 되는 이동 경로가 생성되게 될 것이다. 하지만 이 이동 경로를 최적에 가까운 이동 경로라고 볼 수는 없다.

만약 시작 지점과 목표 지점을 잇는 이상적인 경로상에 놓인 장애물들 사이에 이동 로봇이 통과할 수 있는 공간이 존재하면서도 불구하고 가상으로 그 크기가 확장되어 하나의 형태 장애물만이 존재한다면 해당 형태 장애물을 보다 정확하게 찾아줌으로써 이동 로봇이 통과할 수 있는 공간을 만들어 줄 수 있게 된다. 또한, 이를 통해 보다 최적에 가까

운 이동 경로를 찾아낼 수 있게 된다. 이러한 내용이 그림 2와 그림 3에 나와 있다.

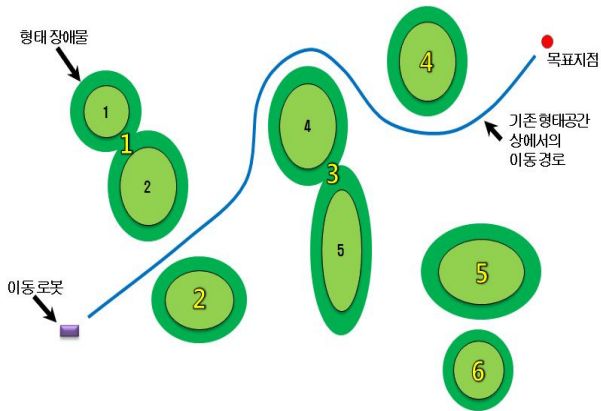


그림 1. 원형 근사화를 통한 이동 경로 계획
Fig 1. Path planning for virtual circular approximation of mobile robot's shape

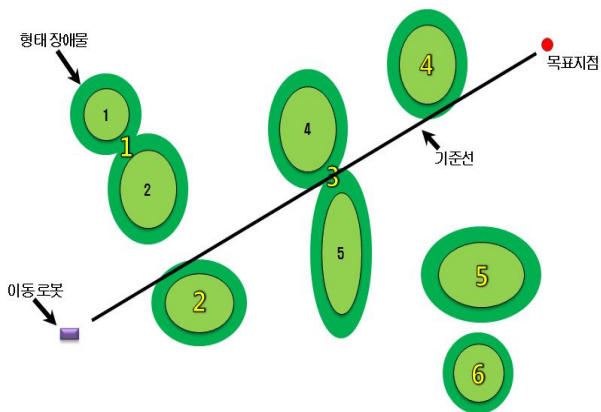


그림 2. 시작 지점과 목표 지점을 잇는 직선
Fig 2. Straight line connecting between the starting point and goal point

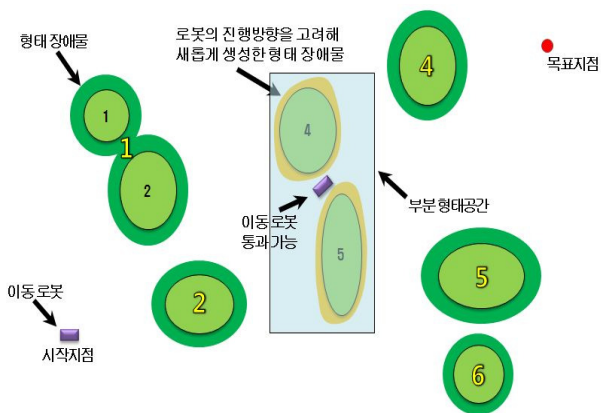


그림 3. 이동 로봇의 형태 및 이동 메커니즘을 고려한 형태 장애물 재생성
Fig 3. Recreate the detailed partial C-space with real shape and movement mechanism of the mobile robot

이동 로봇의 원형 근사화를 통하여 형태 공간을 생성하고 생성된 형태 장애물 중에서 두 개 이상의 구분된 장애물로 구성된 형태 장애물이 있다면 형태 공간을 부분적으로 재구성해야 될 필요성이 있는지 판단하여 본다.

그림 2에서는 이를 판단하는 방법이 나와 있다. 시작 지점과 목표 지점을 잇는 이상적인 최단 경로인 직선을 긋고 이 직선상에 두 개 이상의 실제 장애물로 구성된 형태 장애물이 있는지 판단하여 본다. 만약 직선상에 조건을 만족하는 형태 장애물이 존재할 경우 해당 형태 장애물에 대해서 이동 로봇의 형태 및 이동 메커니즘을 고려하여 새롭게 형태 장애물 및 형태 공간을 재구성하게 된다. 이 내용이 그림 3에 나와 있다.

그림 3은 특정 부분에 한정해서 형태 공간을 보다 상세하게 재생성한 모습이다. 실제 장애물 4번과 5번으로 생성된 형태 장애물이 시작 지점과 목표 지점을 잇는 이상적 경로인 직선 내에 위치하고 있다. 따라서 해당 형태 장애물에 대해서 이동 로봇의 실제 형태 및 이동 메커니즘을 고려하여 형태 장애물을 재구성하게 되면 하나의 형태 장애물이 두 개의 형태 장애물로 나뉘게 되고 이 공간을 이용하여 최적에 가까운 이동 경로를 찾을 수 있게 되는 것이다.

그림 4에서는 형태 공간을 부분적으로 재생성하여 새롭게 구성된 형태 공간을 이용하여 이동 경로 계획을 한 모습이다. 그림 2에서 볼 수 있듯이 이동 로봇을 원형으로 근사화하여 생성한 형태 공간에서 이동 경로 계획을 하였을 때 보다 최적에 가까운 이동 경로를 찾을 수 있는 것을 확인할 수 있다.

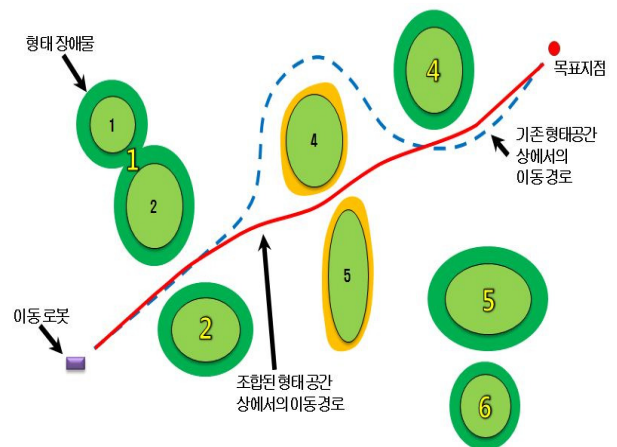


그림 4. 형태 공간에 따른 이동 경로 계획의 차이
Fig 4. Different types of path planning of C-space

본 논문에서는 새롭게 제안한 다단계 형태 공간 생성 방법이 모든 장애물을 대상으로 형태 공간을 정확히 생성해내는 기존의 방법보다 얼마나 효율적인지를 시뮬레이션을 통해 검증하고자 한다. 구체적으로 이동 로봇의 다양한 형태 별로 형태 공간을 생성되는데 소요되는 시간들을 비교, 분석하는 방법으로 성능 검증을 수행하였다.

3. 성능 검증을 위한 실험 환경

새롭게 제안한 다단계 형태 공간 생성 방법에 대한 성능을 검증하기 위하여 Intel i5 750 CPU[2.66GHz], DDR3 4G

메모리, 그리고 MS-Windows 7이 탑재된 PC상에서 Microsoft Visual 2008 C++와 MFC, OpenGL을 이용하여 시뮬레이션을 구현하고 시행하였다.

시뮬레이션을 위한 이동 로봇의 형태로는 그림 4와 같이 총 5개의 형태를 사용하였다.

테스트 진행 환경은 그림 5와 같이 160(cm) × 120(cm) 크기의 평면을 구성하였다. 이평면에 실제 장애물을 20(cm) × 20(cm) 1개, 40(cm) × 20(cm) 1개, 20(cm) × 10(cm) 2개 위치시킨다. 그리고 그림 6에 나와 있는 5개의 이동 로봇 형태를 이용하여 일정한 각도(10°, 1°, 0.5°, 0.1°)별로 이동 로봇을 회전시키면서 이동 로봇의 형태 및 회전각도(heading angle)을 고려하여 모든 장애물들을 대상으로 형태 공간을 생성하였을 경우의 소요 시간과 원형근사화 이후 부분적으로 형태 공간을 생성하였을 경우의 소요 시간을 중앙처리장치(CPU)의 틱 카운트(Tick count)를 이용하여 측정하였다. 단, 중앙처리장치의 틱 카운트를 사용할 경우 PC의 리소스 등 여러 가지 환경적인 요인에 따라서 같은 결과가 나오지 않기 때문에 보다 정확한 실험 결과를 얻기 위하여 모든 실험은 각각 5회씩 수행한 뒤 이들의 평균값을 이용하였다.

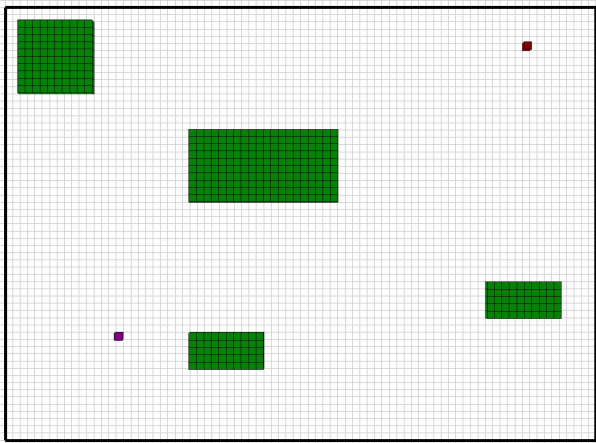
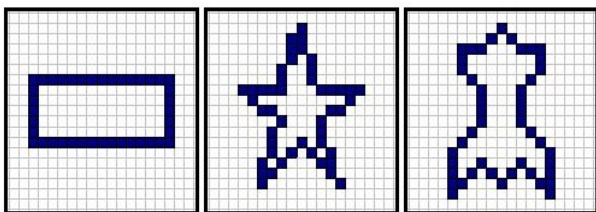
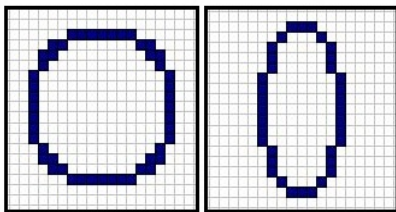


그림 5. 테스트 환경
Fig 5. Environment of Testing



a) 사각형 b) 별 c) 다각형



d) 원 e) 타원

그림 6. 이동 로봇의 형태
Fig 6. Shape of mobile robots

4. 실험 결과

아래의 표 1, 표 2, 표 3, 표 4는 5가지의 이동 로봇의 형태를 이용하여 원형근사화 이후 부분적으로 정확한 형태 공간을 생성(제안한 방법)하는데 걸린 시간과 모든 장애물을 대상으로 형태 공간을 생성(기존의 방법)하는데 걸린 시간을 이동 로봇을 회전시키는 단위각도 별로 정리한 것이다. 각 표의 아래 부분에 삽인 된 시간 단축은 제안된 방법이 기존의 방법에 비해 얼마만큼의 효율을 보이는지를 나타내고 있다.

표 1. 단위회전각도 10°

Table 1. Rotation Degree 10° (단위: 초)

	사각형	별	다각형	원	타원
제안	1.445	2.246	2.683	1.248	0.92
기존	2.018	3.198	3.915	1.248	1.482
시간 단축	28.4%	29.8%	31.5%	5.9%	37.9%

표 2. 단위회전각도 1°

Table 2. Rotation Degree 1° (단위: 초)

	사각형	별	다각형	원	타원
제안	14.758	22.293	26.894	12.324	9.219
기존	20.187	32.74	39.265	12.324	14.726
시간 단축	26.9%	31.9%	31.5%	7.6%	37.4%

표 3. 단위회전각도 0.5°

Table 3. Rotation Degree 0.5° (단위: 초)

	사각형	별	다각형	원	타원
제안	29.656	44.819	54.23	24.773	18.611
기존	40.327	64.85	78.39	24.773	29.468
시간 단축	26.5%	30.9%	30.8%	7.2%	36.8%

표 4. 단위회전각도 0.1°

Table 4. Rotation Degree 0.1° (단위: 초)

	사각형	별	다각형	원	타원
제안	148.185	223.628	269.538	123.724	92.243
기존	202.53	320.473	-	123.724	147.405
시간 단축	26.8%	30.2%	-	7.2%	37.4%

5개의 이동 로봇의 형태를 크게 두 가지로 원형과 비 원형(사각형, 별, 다각형, 타원)으로 나눌 수 있다. 이 같은 분류로 각각의 실험 결과를 살펴보면 이동 로봇의 형태가 원형일 경우보다 비 원형일 경우 형태 공간을 생성하는데 소요되는 시간이 많이 나타나는 것을 확인할 수 있다. 이는 이동 로봇의 형태가 원일 경우에는 이동 로봇의 형태를 원형으로 근사화하는 방법과 마찬가지로 이동 로봇의 회전 자유도를 고려하지 않아도 되기 때문에 형태 공간을 생성하는데 걸리는 소요 시간이 적게 나타나는 것으로 분석된다.

이동 로봇의 형태가 비 원형으로 이루어져 있을 경우도 선분으로 이루어진 이동 로봇의 형태(사각형, 별, 다각형)와

곡선으로 이루어진 이동 로봇의 형태(타원)로 나눌 수 있다. 선분으로 이루어진 이동 로봇들은 평균 29% 정도의 시간 단축률을 보이는 반면 선분이 아닌 형태로 이루어진 경우 37% 정도의 시간 단축률을 보여주고 있다. 이는 선분으로 이루어진 이동 로봇의 경우 해당 형태 장애물을 생성할 때 계산의 간략화를 위해 이동 로봇을 구성하는 각각의 선분들에 대하여 각각 이동 로봇이 회전하는 각도에 맞추어 각도 변환을 해줄 수 있지만, 선분이 아닌 형태(타원)로 이루어진 이동 로봇의 경우 이동 로봇의 형태를 가장 비슷하게 표현할 수 있는 곡선식을 이용하거나 또는 격자지도의 경우 각 픽셀별로 각도 변환을 해주어야 하기 때문이다. 그 결과, 다양한 선분을 가진 다각형의 경우 타원과 같은 곡선형태보다도 계산시간이 많이 걸린 것을 확인할 수 있었다.

표 4를 살펴보면 이동 로봇의 형태가 다각형일 때 모든 장애물을 대상으로 형태 공간을 생성할 때에 대한 실험결과가 없는 것을 볼 수 있다. 이는 해당 케이스로 실험을 진행하였을 경우 메모리 풀(Memory full)이 발생하여 실험결과를 얻지 못하였다. 하지만 그 결과를 전체적인 실험 결과를 토대로 예측하여 보면 390초~400초 정도 소요될 것으로 예상된다.

결론적으로 실험을 통해 새롭게 제안한 다단계 형태 공간 생성 방법이 기존의 방법보다 뛰어난 수행시간 단축효과가 있는 것을 확인하였다.

특히 그림 7은 회전각도 10°에서 부분적으로 형태 공간을 생성하였을 때의 시뮬레이션 화면이다. 그림을 살펴보면 위·아래에 한 줄씩 표현되는 선이 각 각도에 대한 시작 지점과 목표 지점이다. 그림 7의 왼쪽에 있는 두 개의 장애물은 이동 로봇의 모양을 원형 근사화하여 형태 장애물을 생성할 경우 한 개의 형태장애물로 합쳐지게 되고 또한 시작 지점과 목표 지점을 잇는 직선상에 위치하게 된다. 결과적으로 이 경우 보다 효과적인 이동 경로를 찾을 수 있는 가능성이 존재하기 때문에 두 개의 실제 장애물로 나누어지는 것이 바람직하며 그림 7은 제안된 방법에 따라 2개의 실제 장애물로 나누어졌음을 보이고 있다. 즉, 해당 형태 장애물의 경우 이동 로봇의 형태와 이동 메커니즘에 맞게 생성된 것을 확인할 수 있으며 나머지 두 개의 장애물은 이동 로봇의 형태를 원형 근사화하여 생성한 형태 장애물의 모양을 띄고 있다.

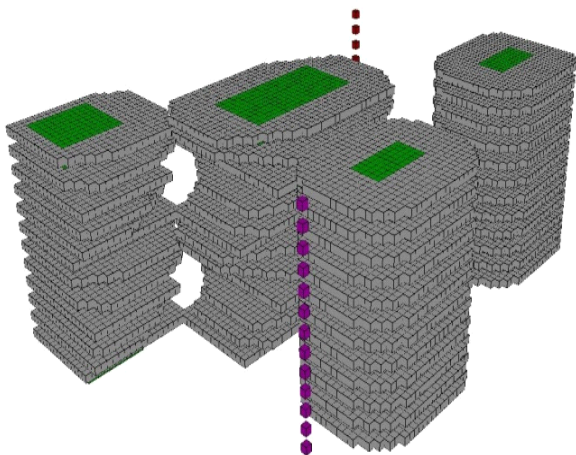


그림 7. 회전각도 10°에서 부분적 형태 공간 생성
Fig 7. Applying 10 degree of rotation for partial obstacles in C-space generation

반면 그림 8은 회전각도 10°에서 모든 장애물을 대상으로 형태 공간을 생성하였을 때의 시뮬레이션 화면이다. 그림 7과 같이 시작 지점과 목표 지점을 잇는 직선상에는 두 개의 실제 장애물들만 존재 하지만 모든 장애물에 대해 이동 로봇의 형태 및 이동 메커니즘을 적용하여 정확한 형태 공간을 구성하였다.

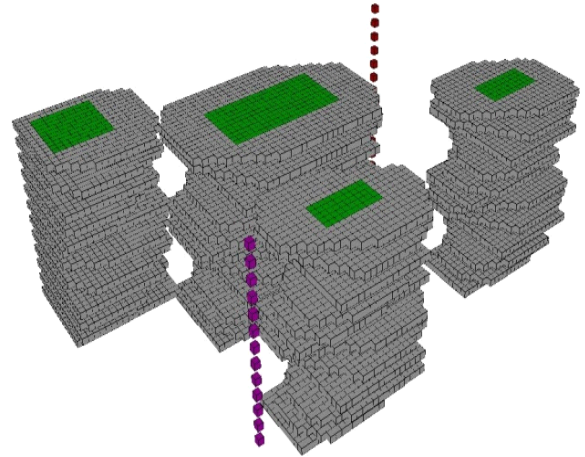


그림 8. 회전각도 10°에서 모든 장애물에 대한 형태 공간 생성

Fig 8. Applying 10 degree of rotation for all obstacles in C-space generation

5. 결 론

본 논문에서는 이동 로봇의 경로 계획을 위한 중요한 개념들 중의 하나인 형태 공간을 효율적으로 구성하는 방법으로 다단계 형태 공간 생성 방법을 제안하고 이에 대한 성능 검증을 실시하였다. 성능 검증을 실시하는 방법으로는 모든 장애물들에 대해 이동 로봇의 형태 및 이동 메커니즘을 적용하여 형태 공간을 생성하였을 경우의 생성 시간과 새롭게 제안한 이동 로봇의 원형 근사화 이후 일부 장애물에만 이동 로봇의 형태 및 이동 메커니즘을 적용하여 정확한 형태 공간을 재생성 하였을 경우의 생성 시간을 비교하였다.

이번 실험에서 사용한 실험 공간은 160(cm) × 120(cm) 크기의 평면에 배치시킨 실제 장애물 4개 중 이동 로봇의 원형 근사화를 통해 두 개의 실제 장애물이 하나의 형태 장애물로 병합되도록 구성하였다. 이러한 환경 하에서 실험을 진행한 결과 이동 로봇의 형태가 원형이 아닐 경우 약 27%~37%의 생성 시간 단축 효과를 확인할 수 있었다. 하지만 이동 로봇의 형태가 원형일 경우 약 5%~7%의 미미한 생성 시간 단축 효율을 보였다.

이동 로봇의 형태가 원형일 경우가 원형이 아닐 경우보다 생성 시간 단축 효율이 낮은 이유는 이동 로봇의 형태가 원형일 경우 이동 로봇의 이동 메커니즘을 고려하지 않아도 되기 때문에 그 차이가 적은 것으로 확인되었다.

만약 이번 실험에서 사용한 환경보다 공간의 크기 및 복잡도가 높아지고 시작 지점과 목표 지점을 연결하는 이상적인 이동 경로인 직선상에 두 개 이상의 실제 장애물로 구성 되어진 형태 장애물의 개수가 많아지게 된다면 이번 실험을 통해 얻은 약 27%~37%의 시간 단축 효율보다 보다 높은 생성 시간 단축률을 보일 것이다.

참 고 문 헌

[1] 정만태, “신 성장 동인으로 주목 받는 로봇산업의 현황과 발전전략”, KEIT 산업경제, 2009년 3월.

[2] T. Lozano-Perez, “A Spatial Planning : A Configuration Space Approach,” *IEEE Transactions on Computers*, Vol. C-32, pp.108-120, 1983

[3] T. Lozano-Perez, “A Simple Motion Planning Algorithm for General Robot Manipulators,” *IEEE Transactions on Robotics and Automation*, Vol. RA-3, no.3, pp.224-238, 1987

[4] Jin-Woo Jung, Byung-Chul So, “Multi-level configuration space for the development of an efficient mobile robot path planner in the embedded system,” *Proc. Of ISET 2010*, Daegu, Korea, May 2010

[5] 정광민, 이희재, 심귀보, “Potential Field를 이용한 자율이동로봇의 경로 계획에 관한 연구,” *한국지능시스템학회 논문지*, vol. 19, no. 5, pp. 737-742, 2009

[6] L.Pimenta and A. R. Fonseca, “Robot Navigation Based on Electrostatic Field Computation,” *IEEE Transaction on Magnetics*, vol. 42, no. 4, April 2006.

[7] R. Daily and D. M. Bevely, “Harmonic Potential Field Path Planning for High Speed Vehicles,” *American Control Conference 2008, Seattle, Washington, USA*, 2008

[8] Thomas Braunl, *Embedded Robotics*, Springer-Verlag, Berlin, Heidelberg, 2003

[9] T. Hesse and T. Sattel, “Path-Planning with Virtual Beams,” *Proc. of the 2007 American Controls Conference*, pp 3904-3905, July 2007

저 자 소 개



박정수(Jung-Soo Park)

2011년 : 동국대학교 컴퓨터공학과(공학사)
2011년~현재 : 동국대학교 컴퓨터공학과 석사과정

관심분야 : 모바일 로봇
E-mail : hostkit@naver.com



소병철(Byung-Chul So)

2010년 : 동국대학교 컴퓨터공학과(공학사)
2010년~현재 : 동국대학교 컴퓨터공학과 석사과정

관심분야 : 모바일 로봇, 인간-로봇 상호작용
E-mail : sbc10620@naver.com



정진우(Jin-Woo Jung)

1997년 : 한국과학기술원 전기 및 전자공학과(공학사)
1999년 : 한국과학기술원 전기 및 전자공학과(공학석사)
2001년~2002년 : 일본 동경대학교 기계정보공학과 대학원 방문연구원

2004년 : 한국과학기술원 인간친화 복지 로봇시스템 연구센터 박사후연구원
2006년~현재 : 동국대학교 컴퓨터공학과 조교수

관심분야 : 인간-로봇 상호작용, 다개체 협력로봇, 소프트웨어 퓨팅, 생체측정, 지능로봇
E-mail : jwjung@dongguk.edu