

## 가상지도를 이용한 청소로봇 경로계획

김형일\*

# Path Planning for Cleaning Robots Using Virtual Map

Hyungil Kim\*

### 요약

가장 많이 활용하는 청소로봇의 청소기법은 크게 랜덤기법과 바둑판식기법으로 나눌 수 있다. 랜덤기법을 이용한 청소로봇은 장애물을 만날 때까지 직진 방향으로 청소를 수행하며, 장애물을 만나면 일정한 각도로 회전한 후 다시 직진 방향으로 청소를 수행한다. 랜덤기법은 중복적인 청소를 수행하는 문제가 빈번히 발생하며, 청소를 완료하는 데 오래 시간이 소요되는 단점이 있다. 바둑판식기법을 이용한 청소로봇은 장애물을 만날 때까지 직진 방향으로 청소를 수행하며, 장애물을 만나면 직진과 회전을 이용하여 청소가 수행되지 않은 위치로 이동하고, 이전에 청소를 수행한 방향의 반대로 청소를 수행한다. 바둑판식기법은 청소공간을 조밀하게 청소하며 진행하기 때문에 장애물이 없거나 작은 공간에서 작업 성능이 뛰어나다. 그러나 바둑판식기법으로 장애물이 있거나 복잡한 공간을 청소할 때는 청소시간이 증가한다. 그러므로 청소의 효율성을 증가시키기 위해서는 작업공간을 정확히 파악하여 청소를 계획적으로 진행해야 한다. 본 논문에서 제안한 가상지도 기반 청소로봇은 작업공간을 효율적으로 청소하는 특성을 갖는다. 시뮬레이션을 통해 제안한 기법의 효율성을 측정하였다.

### Abstract

The most common cleaning methods of cleaning robot can be divided into two categories: the random and the boustrophedon method. A cleaning robot using the random method moves straight until it bumps into an obstacle. If it collides with an obstacle, the cleaning robot turns a specific angle and continues moving straight. Therefore, the random method often tends to clean the already clean area repeatedly. In addition, it takes a long time to complete cleaning. A cleaning robot using the boustrophedon method moves straight until it collides with an obstacle. If it meets an obstacle, the cleaning robot moves to the next uncleaned space through turning and moving ahead. when resuming cleaning from the new region, a cleaning robot moves in the direction opposite to the direction of the previous cleaning. Because the boustrophedon method cleans a cleaning space more densely, its performance is excellent in an obstacle-free space or a small space. However, In a space with obstacles or a complex structure, it takes a long time to complete the cleaning work. Cleaning should be systematically approached with a good understanding of the

---

\*제1저자 : 김형일  
\*투고일 : 2009. 11. 06, 심사일 : 2009. 11. 18, 게재확정일 : 2009. 11. 20.  
\* 나사렛대학교 멀티미디어학과 교수

work area. The virtual map-based cleaning robot proposed in this paper cleaned a work space efficiently. The efficiency of the proposed method was measured through simulation.

▶ Keyword : 청소로봇(Cleaning Robot), 가상지도(Virtual Map), 경로계획(Path Planning), 청소기법(Cleaning Method)

## 1. 서론

현재 로봇 연구는 인간이 직접 수행하기 힘든 작업들을 대신하거나 인간에게 편의를 제공하는 목적의 서비스 로봇 분야에서 활발히 진행되고 있다. 서비스 로봇은 수리, 이송, 청소 및 건강관리 등과 같은 일을 주로 수행하며, 서비스 로봇은 크게 인간 서비스 로봇과 홈 서비스 로봇으로 분류된다[1, 2].

서비스 로봇 중에서 가장 먼저 상용화 된 제품은 청소로봇이다[3, 4, 5]. 청소로봇의 상업적 활성화 이유는 가사 부담을 줄여줄 뿐만 아니라, 노동이나 움직임이 불편한 장애인 및 노인의 복지 문제를 해결할 수 있기 때문이다. 현재 시판되고 있는 청소로봇에서 주로 활용하는 청소기법은 랜덤기법(random method)과 바둑판식기법(boustrophedon method)이다[6].

랜덤기법은 직진 방향으로 청소를 수행하다 장애물을 만나면 일정한 각도로 회전한 후 다시 직진 방향으로 청소를 수행하는 방식을 취한다[7]. 랜덤기법은 단순한 규칙에 따라 청소를 수행하기 때문에 청소로봇의 구조가 간단하다. 랜덤기법으로 청소를 수행할 경우 중복적인 청소영역이 많이 발생할 뿐만 아니라 청소가 수행되지 않은 영역이 빈번하게 발생한다. 그리고 청소를 완료하는 데 오랜 시간이 걸린다. 랜덤기법은 좁은 작업공간에서는 어느 정도 효과를 발휘하지만, 작업공간이 넓은 경우에는 청소효과가 떨어지는 문제점이 있다. 그림 1에 랜덤기법으로 청소를 수행하는 방법을 표현하였다.

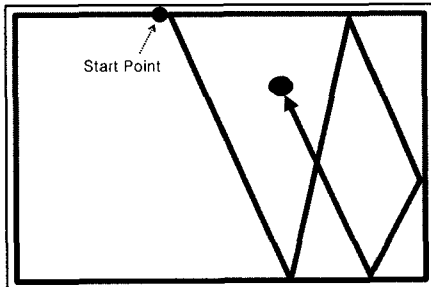


그림 1. 랜덤기법  
Fig. 1. The random method

바둑판식기법은 랜덤기법 외에 가장 많이 활용된 기법으로 랜덤기법에 비해 중복 청소영역이 적고, 세밀하게 작업공간을 청소할 수 있다[8]. 그러나 바둑판식기법은 장애물이 존재하는 환경에서는 한 번에 모든 작업공간을 청소하기 힘든 경우가 많으며, 중복적인 청소영역도 빈번하게 발생한다[9, 10, 11]. 그림 2에 바둑판식기법으로 청소를 수행하는 방법을 표현하였다.

이와 같은 청소로봇의 문제점을 해결하기 위해 본 논문에서는 가상지도를 이용한 청소로봇의 효과적인 경로계획기법을 제안한다. 제안 기법이 사용된 청소로봇은 환경탐색에 의해 가상지도를 생성하고, 생성된 가상지도는 청소를 수행할 때 활용된다. 본 논문에서 제안한 가상지도를 이용한 경로계획기법은 공간 복잡도가 높은 작업공간에서도 효과적인 청소를 수행할 수 있다.

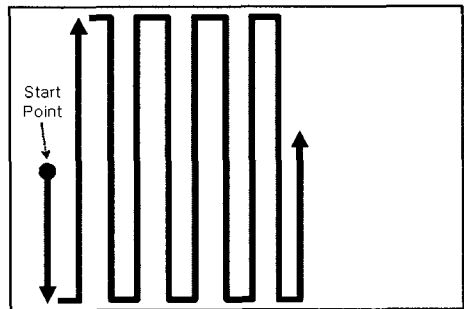


그림 2. 바둑판식기법  
Fig. 2. The boustrophedon method

## II. 관련 연구

Ribes는 랜덤기법의 방향전환 각도를 80도에서 135도 범위에서 랜덤하게 선택하여 청소의 효율성을 높였다[12]. 그러나 랜덤기법은 기존의 여러 가지 문제점으로 청소 효율성이 떨어지고, 에너지 낭비가 심하다.

Rekleitis 등은 바둑판식기법의 문제점인 중복 청소영역을 최소화시키기 위해 여러 청소로봇이 하나의 청소공간을 바둑판식으로 청소하는 기법을 연구하였다[13]. 청소로봇들은

음파탐지기를 이용하여 청소가 수행된 공간과 수행되지 않은 공간을 확인하며 청소를 수행한다.

이와 같이 여러 청소로봇을 이용한 다른 연구로 RFID 기반 청소로봇 연구가 있다. 하나의 청소로봇만을 이용할 경우에는 중복 청소영역이 많이 발생하며, 이러한 문제를 해결하기 위해 여러 청소로봇의 협력작업을 이용하여 청소를 수행하는 연구가 시도되었다[14]. 이 연구는 RFID 태그를 이용하여 청소로봇의 위치를 찾아 중복 청소영역을 최소화하며 청소를 수행하는 방법에 대한 연구이다.

경로계획은 자유롭게 이동할 수 있는 이동로봇이 목적하는 위치로 이동하기 위해 자신의 상태와 주변정보를 기반으로 목표지점까지 경로를 계획하는 방법이다[15]. 이러한 경로계획에서 전체 영역 경로계획은 작업공간 전체에 대한 경로계획을 수립하는 방법이다[16]. 전체 영역 경로계획에서 활용되는 방법에는 템플릿(template)과 포텐셜필드(potential field) 등이 있다[17, 18, 19].

템플릿방식은 청소로봇이 수행할 동작들을 템플릿으로 지정하고, 템플릿에 따라서 작업공간을 청소하는 방법이다. 템플릿에는 청소로봇이 수행해야 할 동작들이 순서대로 나열되어 있으며, 청소로봇은 이러한 템플릿의 명령에 따라 작업공간을 청소하며 이동하게 된다.

Hofner 등은 유턴 템플릿과 쉬프트-사이드 템플릿을 정의하여 청소를 수행하는 방법에 대해 연구하였다[20]. 유턴 템플릿은 청소로봇이 작업공간에서 장애물을 만났을 때 장애물에 방해받지 않는 방향으로 180도 회전하는 템플릿이다. 쉬프트-사이드 템플릿은 청소로봇이 청소를 수행하며 옆 공간으로 비스듬히 이동하는 템플릿이다.

Carvalho 등은 다섯 가지 템플릿을 활용하여 청소를 수행하는 방법에 대해 연구하였다[8]. 다섯 가지 템플릿은 직진 템플릿, 유턴 템플릿, 쉬프트-사이드 템플릿, 연속형 유턴 템플릿, 백트래킹 템플릿이다.

포텐셜필드방식은 청소영역을 격자 형태로 표현하며, 시작 지점부터 목표지점까지의 모든 영역에 대해 거리값을 할당한다. 청소로봇은 자신이 방문하지 않은 인접한 지점들 중에서 거리값이 가장 큰 지점으로 이동하며 청소를 수행한다[17].

Zelinsky 등은 포텐셜필드에서 각 지점에 할당하는 거리값에 변화를 주었으며, 전체 영역 경로를 위한 포텐셜필드뿐만 아니라 목적지까지의 최단경로를 찾기 위한 포텐셜필드의 거리값 계산도 제시하였다[19]. 이러한 포텐셜필드는 전체 작업공간과 장애물에 대한 모든 환경정보를 가지고 있어야 계산을 수행할 수 있다.

표 1. 랜덤기법과 비독판식기법 비교  
Table 1. Comparison between the random method and the boustrophedon method

종류	싱글 로봇		멀티 로봇
	랜덤기법	비독판식기법	비독판식기법
완료도	낮음	높음	매우 높음
중복도	매우 높음	높음	낮음
비용	매우 낮음	낮음	높음

표 2. 경로계획  
Table 2. Path planning

종류	템플릿	포텐셜필드
범위	전역 경로	전역 경로
동작정의	정의함	정의하지 않음
거리값	사용하지 않음	사용함

본 논문에서는 싱글 로봇을 활용하여 비용을 낮추면서 랜덤기법과 비독판식기법에 비해 청소시간을 빠르게 하는 방법에 대해 제안한다. 제안한 외곽순회기법은 전역 경로계획에서 행동해야 할 규칙을 미리 정의하지 않아도 모든 작업영역을 청소할 수 있다는 장점이 있다.

### III. 가상지도를 이용한 청소기법

본 논문에서 제안한 가상지도를 이용한 외곽순회기법은 청소로봇을 통해 환경탐색을 수행하여 작업공간에 대한 가상지도를 생성하고, 생성한 가상지도를 이용하여 청소를 수행한다. 과거에 청소를 수행한 작업공간에서 청소를 진행할 경우에는 기존 가상지도에서 청소로봇 위치를 추출하여 청소를 진행하는 방식을 따른다.

#### 3.1 환경탐색과 가상지도 생성

집이나 사무실 등의 환경에서 청소를 수행할 때 가장 정적인 객체는 벽면이다. 이와 같은 작업환경을 고려하여 청소로봇은 환경의 크기를 탐색하기 위하여 벽면순회를 시행하며, 벽면순회를 통해 작업공간의 크기와 윤곽을 알아낸다. 이러한 과정을 진행할 때 청소로봇은 가상지도를 작성하며 청소를 진행한다.

본 논문에서 제안한 방법에 의해 생성된 가상지도는 다음과 같은 정의에 의해 완전하다. 가상지도를 구성하는 각 격자는 셀이라 정의하고, 가상지도 좌표계는 cartesian product에 의해 생성할 수 있다.  $n$ 개의 집합  $X_1, X_2, \dots, X_n$ 의 cartesian product를 식 1에 표현하였다.

$$X_1 \times X_2 \times \dots \times X_i \times \dots \times X_n = \{(x_1, x_2, \dots, x_i, \dots, x_n) | x_i \in X_i, i \in N\} \quad (1)$$

이러한 전제에 의해 현재 처한 전체공간을  $X$ 라 가정하고, 전체공간  $X$ 에서 외곽 장애물(벽면)을 기준으로 내부공간을  $U$ 라 정의하자.  $X$ 가 거리함수  $d$ 를 가진 집합이라 할 때,  $U$  내의 임의의  $x$ 에 대해  $x$ 의 적당한 근방이  $U$ 의 부분집합으로 존재하면  $X$ 의 부분집합  $U$ 를 open set이라 한다.  $X$ 가 임의의 집합이고  $R$ 을 실수 전체집합이라 할 때  $d$ 를  $X$ 상의 함수라 하자. 그리고  $X$ 상의 함수  $d$ 가 식 2로 표현될 때 임의의  $x, y, z$ 에 대하여 식 3, 식 4, 식 5, 식 6을 만족하면  $d$ 를 집합  $X$ 상의 거리함수라 한다. 이 때 거리함수  $d$ 를 가진 집합  $X$ 를 거리공간이라 하고  $(X, d)$ 로 표현한다.

$$d: X \times X \rightarrow R \quad (2)$$

$$d(x, y) \geq 0, x, y \in X \quad (3)$$

$$d(x, y) = d(y, x), x, y \in X \quad (4)$$

$$d(x, y) = 0 \text{ iff } y = x, x, y \in X \quad (5)$$

$$d(x, y) + d(y, z) \geq d(x, z), x, y, z \in X \quad (6)$$

청소로봇이 처한 공간을  $X$ 로 정의하고 청소공간을  $U$ 로 정의할 때 전체공간  $X$ 상의 거리함수  $d$ 를 유클리디안 거리(Euclidean distance)로 정의할 수 있다.  $X$ 상에  $x$ 와  $y$ 가 존재할 때 유클리디안 거리  $d(x, y)$ 는 식 7로 표현된다. 그러면 청소공간  $U$ 는 open set으로 정의되어 모든 영역을 완전한 청소 영역으로 확보할 수 있으므로 open set  $U$ 에 속한 모든 셀들을 청소로봇이 방문할 수 있다. 그러므로 청소로봇이 활용할 가상지도는 거리함수를 소유한 청소에 완전한 공간이다.

$$d(x, y) = \sqrt{\sum_{i=1}^n |x_i - y_i|^2} \text{ for } x_i, y_i \in X, i \in N \quad (7)$$

청소로봇이 환경을 탐색할 때 환경정보는 청소로봇의 이동형태에 따라 진행정보로 표현되어 가상지도에 저장된다. 가상지도 생성을 위해 정의한 청소로봇의 이동형태는 직진, 우회전, 좌회전이다. 세 가지 이동형태는 가상지도에서 직진은 "F"로 표현되고, 우회전은 "R"로 표현되며, 좌회전은 "L"로 표현된다. 그림 3는 가상지도와 청소로봇의 이동정보를 표현한 그림이다. 그림에서 화살표는 청소로봇이 이동한 경로를 표현한 것이고, 청소로봇의 이동형태에 따른 이동정보를 가상지도 아래에 표현하였다.

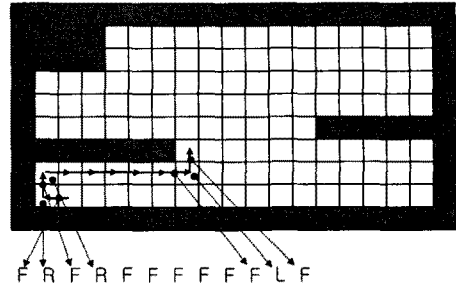


그림 3. 가상지도와 청소로봇의 이동정보  
Fig. 3. Virtual map and information on the movement of cleaning robots

### 3.2 가상지도 추출

현재 청소로봇이 존재하는 작업환경이 이전에 작업했던 환경인 경우에는 가상지도 생성할 필요가 없다. 이러한 경우 저장되어 있는 가상지도 추출하여 현재 작업환경에 대한 가상지도로 활용한다. 가상지도 추출을 위해 초기 환경탐색에서 얻어진 이동정보를 활용하여 가상지도 후보집합을 생성하고, 청소로봇은 현재의 작업환경을 계속적으로 탐색하면서 가상지도 후보집합에서 적합한 가상지도 추출한다. 청소로봇의 이동정보와 가상지도에 나타난 이동정보를 이용하여 현재 작업공간에 적합한 가상지도 추출할 수 있다. 가상지도 추출 단계를 통하여 현재 작업환경이 새로운 작업환경인지 탐색이 수행되었던 작업환경인지 파악할 수 있다.

가상지도 추출과정은 다음과 같다. 먼저 청소로봇은 현재 위치에서 임의의 방향으로 벽면을 만날 때까지 이동한다. 벽면 장애물을 만나면 청소로봇은 해당 지점을 시작지점으로 하여 시계방향으로 벽면순회를 진행하며 청소로봇의 이동형태를 정보화한다. 청소로봇이 이동할 때 나타난 시작지점에서 임의의 종료지점까지의 이동정보를 이동 윈도우라 정의한다. 일정 부분 순회하며 얻어진 청소로봇의 이동 윈도우를 가상지도에 나타난 이동정보와 비교하여 가상지도 추출한다.

윈도우 크기는 세 가지 방법으로 결정할 수 있다. 하나는 일정한 셀의 크기를 윈도우 크기로 결정하는 방법이고, 다른 하나는 윈도우 크기를 셀 단위로 하나씩 증가시켜 얻어진 경로정보를 윈도우 크기로 결정하는 방법이다. 세 번째 방법은 지역 시작지점과 지역 종료지점을 윈도우 크기로 결정하는 방법이다.

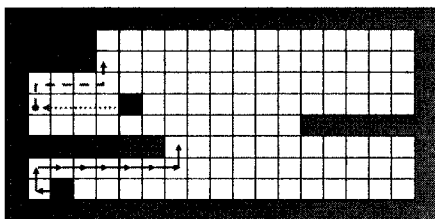
장애물 벽면의 시작지점을 측면 시작지점이라 정의하고, 방향 전환이 이루어진 지점을 측면 종료지점이라 정의한다. 임의의 시작지점을 지역 시작지점이라 하고, 임의의 종료지점을 지역 종료지점이라 정의한다. 청소로봇이 처음 만난 벽면 부분이 지역 시작지점이 되고 시계방향으로 진행하여 처음 방

향 전환이 발생한 지점이 지역 종료지점이 된다. 처음 발생한 지역 시작지점에서 지역 종료지점까지가 최초의 이동 윈도우가 된다.

이렇게 생성된 이동 윈도우와 가상지도의 이동정보를 비교하여 가상지도를 추출한다. 가상지도 추출이 종료되지 않았다면 시계방향으로 계속적인 탐색을 수행하다 방향전환이 발생된 지점이 두 번째 지역 종료지점이 되고, 최초의 지역 종료지점은 두 번째 지역 시작지점이 된다. 새롭게 생성된 두 번째 지역 시작지점과 지역 종료지점으로 두 번째 이동 윈도우로 결정하고 가상지도에 나타난 이동정보와 비교한다. 이와 같은 방법으로 계속적인 이동정보 비교를 수행하여 과거에 환경탐색이 이루어진 가상지도를 추출한다. 가상지도 추출이 종료되는 경우는 현재 작업환경에 대한 전체 벽면순회가 종료되는 시점이다. 이러한 경우는 환경탐색에서 생성된 이동 윈도우와 가상지도에 나타난 이동정보를 모두 비교해 본 경우이다.

청소로봇의 위치 추출은 작업환경과 일치하는 가상지도 추출과 동시에 이루어진다. 현재 작업환경에서 발생한 이동 윈도우가 가상지도 후보집합에 속한 가상지도들에서 나타나지 않으면 청소로봇은 현재 작업환경을 새로운 작업공간으로 정의한다.

그림 4는 가상지도에 나타난 이동정보와 이동 윈도우를 비교하는 간단한 방법을 예제로 나타낸 그림이다. 그림 아래 나타난 청소로봇의 이동정보를 가상지도에 나타난 이동정보와 비교하여 가상지도를 추출한다. 사각형 실선에 나타난 "FRFFFLF"가 이동 윈도우 크기가 된다.



FRFRFFFFFFFFLFFLFFFFFFFFRFRFFFLFRFFFF...  
FRFFFLF  
FRFFFLF  
그림 4. 이동 윈도우와 이동정보 비교  
Fig. 4. Comparison between the moving window and moving information

### 3.3 경로계획

경로계획은 가상지도가 생성되거나 현재 작업환경과 일치하는 가상지도를 찾았을 때 수행되며, 가상지도를 활용하여 작업공간을 효과적으로 청소할 수 있는 이동경로를 생성하는 방법이다.

경로계획을 위한 전제는 다음과 같다. 가상지도에 나타난

장애물로 이용되는 외곽 측면에 가장 근접한 한 변에 속한 셀들의 집합을 변집합이라 정의한다. 각 변집합은 전체 작업공간  $X$ 에 포함되는 부분집합이다. 정의의 집합  $X$ 의 임의의 원소  $x, y, z$ 가 식 8, 식 9, 식 10을 만족할 때 집합  $X$ 를 순서집합이라 한다. 순서집합에서는 관계가 정의될 수 있으며, 식 8은 반사관계, 식 9는 대칭관계, 식 10은 추이관계이다.

$$x \leq x, x \in X \tag{8}$$

$$x \leq y, y \leq x \Rightarrow x = y, x, y \in X \tag{9}$$

$$x \leq y, y \leq z \Rightarrow x \leq z, x, y, z \in X \tag{10}$$

임의의 변집합을  $X_i$ 라 정의하면  $X_i$ 는 집합  $X$ 의 부분집합이고 순서집합이다. 이 때  $X_i$ 에  $a$ 가 존재하고  $a$ 가 식 11을 만족하면  $a$ 를  $X_i$ 의 상계라 하고,  $X_i$ 에  $b$ 가 존재하고  $b$ 가 식 12를 만족하면  $b$ 를  $X_i$ 의 하계라 한다.

$$x \leq a \text{ for } \forall x \in X_i, i \in N \tag{11}$$

$$x \geq b \text{ for } \forall x \in X_i, i \in N \tag{12}$$

청소로봇의 작업공간을 집합  $X$ 에 대응시킬 때 작업공간은 순서집합으로 정의되며, 이동의 기본단위인 셀들은 관계를 형성한다. 그러므로 각 변집합들은 순서집합이 되어 청소로봇이 각 셀을 접근할 때는 크기 비교에 따라 순서적으로 접근할 수 있다.

청소로봇이 처음 만난 외곽 측면을 일차일변집합이라 정의하고, 두 번째, 세 번째, 네 번째 만난 외곽 측면을 각각 일차이변집합, 일차삼변집합, 일차사변집합으로 정의한다. 일차일변집합에 나타난 상계와 하계를 각각 일차일변상계와 일차일변하계라 정의한다. 그러므로 각 변집합에는 최소 하나의 상계와 하나의 하계가 존재한다. 일차일변부터 일차사변까지를 합한 집합을 일차변집합이라 정의한다. 일차변집합에 바로 인접한 내부 변집합을 이차변집합이라 정의한다. 이차변집합에도 상계와 하계가 존재하며, 각각 이차상계와 이차하계로 정의한다. 변집합은 n차변집합까지 확장 가능하다.

청소로봇이 작업을 진행할 때 탐색되는 셀집합들에서 청소가 수행되지 않은 셀집합을 열린 셀집합이라 정의하고, 청소가 수행된 셀집합을 닫힌 셀집합이라 정의한다. 그리고 동일 측면에 속한 셀들의 모임을 측면 셀집합이라 정의한다. 이러한 셀집합들에 속한 셀들은 청소로봇 작업의 최소 단위가 된다.

경로계획은 전체를 위배하지 않는 조건 아래 다음과 같은 단계를 따르며, 탐색은 일정한 방향(예:시계방향)으로 진행된다. 좌에서 우로 탐색을 진행할 때 각 변집합의 상계가 청소

로봇의 목표지점이 되며, 위에서 좌로 탐색을 진행할 때 각 변집합의 하계가 목표지점이 된다. 아래에서 위로 탐색을 진행할 때는 상계가 목표지점이 되며, 위에서 아래로 탐색을 진행할 때는 하계가 목표지점이 된다. 수평으로 이동할 때 목표지점 탐색범위는 cartesian product에 의해 표현된 순서쌍 첫 번째 원소들의 시퀀스(sequence)가 되며, 수직으로 이동할 때 목표지점 탐색범위는 순서쌍 두 번째 원소들의 시퀀스가 된다. 탐색은 청소와 같이 수행된다.

외곽선순회기법을 적용한 청소로봇의 청소 진행은 크게 10 단계로 나뉘며, 청소를 수행하는 과정을 흐름도로 작성하여 그림 5에 나타내었다.

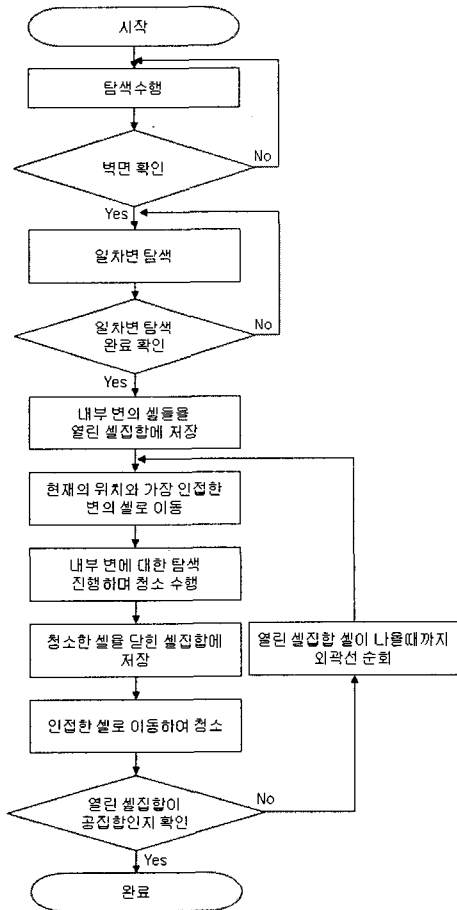


그림 5. 흐름도  
Fig. 5. Flow chart

단계 1  
벽면을 만날 때까지 직진하며 탐색을 수행한다.

단계 2  
벽면을 만나면 일차변 탐색을 수행한다.

단계 3  
일차변 탐색을 진행하면서 방향전환이 수행될 때 상황에 맞는 상계와 하계를 생성한다.

단계 4  
일차변 탐색이 완료되면 내부 변에 대한 상계와 하계를 생성한다.

단계 5  
내부 변들에 속한 모든 셀들을 열린 셀집합에 저장한다.

단계 6  
내부 변들에서 현재의 위치와 가장 인접한 변의 셀로 이동한다.

단계 7  
현재 속한 내부 변에 대한 탐색을 진행하고, 탐색을 진행할 때는 상계와 하계를 이용하여 청소를 수행한다.

단계 8  
청소가 수행된 셀은 닫힌 셀집합에 저장하고, 상계와 하계를 이용하여 인접한 셀로 이동한다.

단계 9  
이동한 셀이 닫힌 셀집합에 속하고 열린 셀집합이 공집합이 아니라면 열린 셀집합에 속한 셀이 나올 때까지 외곽선 순회를 진행한다.

단계 10  
단계 6에서 단계 9를 반복 수행하면서 열린 셀집합이 공집합이 될 때까지 청소를 진행한다.

#### IV. 시뮬레이션

시뮬레이션은 CPU 2.4GHz, 주기억장치 2GB, 윈도우즈 XP 환경에서 Visual C++6.0을 활용하여 구현하였다. 시뮬레이션을 위해 서로 다른 구조로 이루어진 세 개의 가상한

경을 생성하고, 서로 다른 특성을 소유한 네 개의 청소로봇을 구현하였다. 시뮬레이션에서는 청소로봇의 성능을 수행시간으로 측정하였다.

시뮬레이션을 위해 생성한 작업환경은 총 세 가지이며, 모든 작업환경은 1500cm x 1500cm 크기로 제한하였다. 작업환경의 복잡도에 따라 작업환경은 환경I과 환경II 그리고 환경III으로 나뉜다.

환경I은 기본적으로 직사각형 형태를 유지한 상태에서 환경을 랜덤하게 생성하였으며, 세 가지의 작업환경들에서 가장 단순한 작업환경이다. 작업환경의 기본 골격은 직사각형 구조이며, 벽면에 장애물이 나타나지 않는 환경이다. 환경II는 작업환경 외곽에 벽면 장애물이 2개가 나타난 환경으로 환경I에 비해 복잡한 구조를 이루고 있다. 환경II에는 두 개의 벽면에 각각 하나의 장애물이 존재한다. 기본적인 작업환경은 직사각형 구조이나, 작업환경 외곽에 장애물 구조가 결합된 환경이다. 환경III은 작업환경 벽면에 장애물이 나타나는 환경으로 세 가지 작업환경에서 가장 복잡한 환경이다. 환경III에서는 네 개의 벽면에 각각 하나의 장애물이 존재하며, 기본적인 작업환경은 직사각형 구조로 이루어졌다. 각 환경에서 장애물을 생성할 때는 두 가지 전제를 기준으로 랜덤하게 생성하였다. 하나의 전제는 각 장애물은 서로 인접할 수 없다는 것이며, 다른 하나의 전제는 장애물은 하나의 벽면에만 인접하여 생성될 수 있다는 것이다.

시뮬레이션에 사용한 청소로봇은 R1, R2, R3, R4로 네 가지 종류 모두 서로 다른 특성을 갖는다. 설계한 청소로봇의 크기는 15cm로 제한하였으며, 이동속도는 15cm/sec로 한정하였다. 청소로봇의 이동형태는 전진, 우회전, 좌회전만 가능하도록 하여 상용화 청소로봇의 특성에 따라 설계하였다.

청소로봇 R1은 가장 단순한 형태로 작업환경에 대한 정보가 제공되지 않으며, 랜덤기법으로 청소를 수행한다. R1은 직진으로 청소를 수행하며, 장애물을 만났을 경우에는 일정한 각도로 방향을 전환한 후 직진으로 청소를 수행하는 청소로봇이다. 청소로봇 R2는 작업환경에 대한 정보가 없으며, 바둑판식기법으로 청소를 수행한다. 장애물을 만나면 무조건 한 셀 옆으로 이동하여 이전 청소 방향의 반대로 청소를 수행하는 청소로봇이다. 공간에 대한 정보가 없으므로 지속적으로 청소를 반복하는 방식으로 청소를 수행한다. 청소로봇 R3는 벽면순회를 통하여 공간을 탐색하고 바둑판식기법으로 청소를 수행하며, 공간탐색을 통해 가상지도를 생성한다. 일정 방향으로 청소를 수행하고 청소가 안 된 공간이 존재하면 그 공간으로 이동하여 청소를 수행한다. 모든 청소 및 이동에는 바둑판식기법이 적용된다. 과거에 탐색이 이루어진 공간인 경우에는

가상지도를 추출하여 자신의 위치를 찾아낸 후에 청소를 진행한다. 청소로봇 R4는 환경을 탐색하여 가상지도를 생성하는 기능을 갖는다. 청소로봇 R4는 외곽선 순회를 통해 공간을 탐색하고 청소를 수행하며, 모든 이동과 청소는 외곽선 순회기법을 이용한다. 과거에 청소를 수행했던 지역인 경우에는 가상지도를 추출하여 자신의 위치를 찾아낸 후에 청소작업을 수행한다.

그림 6은 환경I에 대한 청소로봇들의 시뮬레이션 결과이다. 환경I은 작업환경의 벽면에 장애물이 존재하지 않는 단순한 형태의 작업환경이다.

R1은 랜덤기법이 적용된 청소로봇으로 벽면순회와 환경탐색을 수행하지 않는 청소로봇이다. R1은 모든 시뮬레이션 환경에서 청소를 완료하지 못하였다. 랜덤기법은 청소를 효율적으로 수행할 수 없다는 것을 본 시뮬레이션으로 확인할 수 있었다. R2는 시뮬레이션 환경I에서 6,271의 성능을 나타냈다. R2는 바둑판식기법이 적용된 청소로봇으로 벽면순회와 환경탐색을 수행하지 않는다. R1이 청소를 완료하지 못한 것에 비해 R2는 모든 시뮬레이션 환경에서 청소를 완료하여 바둑판식기법이 랜덤기법보다 청소 효율이 높음을 본 시뮬레이션을 통해 확인할 수 있었다. R3는 6,134의 성능을 나타내었다. R3는 바둑판식기법으로 청소를 수행하는 청소로봇으로 환경탐색을 통해 가상지도를 생성하기 때문에 작업공간을 인식하여 R2에 비해 우수한 성능을 나타내었다. R4는 환경I에서 5,923 성능을 나타내어 가장 우수한 결과를 나타내었다. R4는 환경탐색을 통해 가상지도를 생성하고, 외곽선순회기법으로 청소를 수행하는 청소로봇이다. 이와 같은 특성으로 R4는 R3에 비해 우수한 성능을 나타내었다.

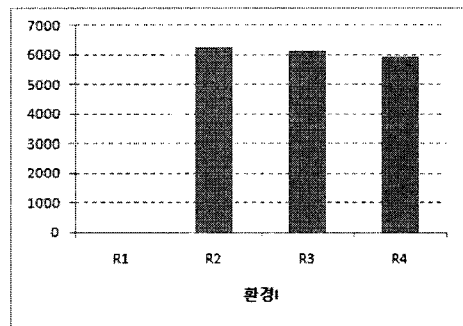


그림 6. 청소시간(환경I)  
Fig. 6. Cleaning time(environment I)

그림 7은 환경II에 대한 청소로봇의 시뮬레이션 결과이다. 환경II는 작업환경의 벽면에 장애물이 두 개가 존재하는 구조

이다. 환경II는 환경I에 비해 복잡한 구조로 이루어진 환경으로 환경I에 비해 효율적인 청소가 이루어지지 않는 환경이다.

환경II에서 R2는 8,945의 성능을 나타내어 환경I에서 나타난 R2 성능에 비해 43%의 성능 저하가 발생하였다. 이러한 시뮬레이션 결과가 발생한 이유는 복잡한 환경적 구조에서 찾을 수 있다. R3는 7,229의 성능을 나타내어 환경I에 비해 17.85%의 성능 저하가 발생하였다. 그러나 R3가 R2에 비해 우수한 성능을 나타낸 이유는 R3는 벽면순회를 통해 작업 공간을 탐색했기 때문이다. R4는 6,528의 성능을 나타내어 환경I에 비해 10.21%의 성능 저하가 발생하였다. 그러나 R4는 다른 청소로봇에 비해 가장 우수한 청소 성능을 나타내었다.

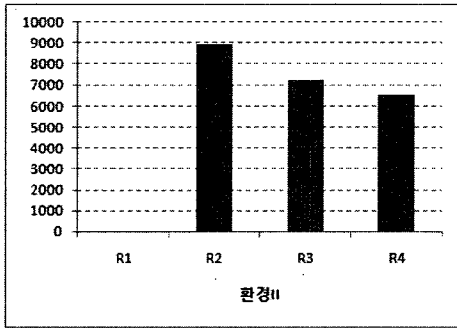


그림 7. 청소시간(환경II)  
Fig. 7. Cleaning time(environment II)

그림 8은 환경III에 대한 청소로봇의 시뮬레이션 결과이다. 환경III은 작업환경의 벽면에 장애물이 네 개 존재하는 구조이다. 환경III은 환경I과 환경II에 비해 복잡한 구조를 가지며, 복잡한 구조는 청소의 효율성을 저하시키는 중요한 환경적 요소이다.

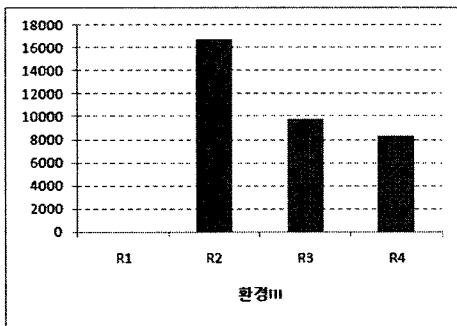


그림 8. 청소시간(환경III)  
Fig. 8. Cleaning time(environment III)

환경III에서 R2는 16,743의 성능을 나타내어 환경II에서 나타난 성능에 비해 87%의 성능 감소가 나타났다. 이와 같이

성능 감소가 크게 나타난 이유는 환경III은 환경I과 환경II에 비해 공간 복잡도가 높기 때문이다. R2는 환경탐색을 수행하지 않기 때문에 복잡한 환경에서는 효율적인 청소를 수행할 수 없다는 것을 본 시뮬레이션을 통해 확인하였다. R3는 9,835의 성능을 나타내어 환경II에서 나타나 성능에 비해 36%의 성능 감소가 나타났다. R2에 비해 R3가 우수한 성능을 나타낸 이유는 R3는 환경탐색을 수행하며 가상지도를 생성하기 때문이다. R4는 8,326의 성능을 나타내어 환경II의 성능에 비해 27%의 성능 감소가 나타났다. R4가 가장 우수한 성능을 나타낸 이유는 탐색을 통해 가상지도를 생성하고, 외곽선순회 기법을 이용하여 효율적인 청소를 수행했기 때문이다.

그림 9는 모든 환경에 대한 청소로봇들의 시뮬레이션 결과로 R2는 10,653의 평균 성능을 나타내었고, R3는 7,733의 평균 성능을 나타내어 R2에 비해 27.41%의 성능 향상을 나타내었다. R3는 가상지도 생성과 환경탐색을 수행하여 R2에 비해 우수한 시뮬레이션 결과를 나타낸 것이다. R4는 6,926의 평균 성능을 나타내어 R3에 비해 10.44%의 성능 향상을 나타내었고, R2에 비해 34.99%의 성능 향상을 나타내었다. R4는 탐색을 통해 가상지도를 생성하고 외곽선순회기법을 활용하기 때문에 R3에 비해 우수한 청소 성능을 나타낸 것이다.

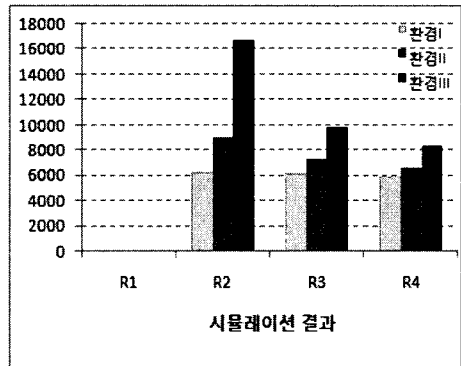


그림 9. 시뮬레이션 결과  
Fig. 9. Simulation result

## V. 결론

가장 많이 활용하는 청소로봇의 청소기법은 랜덤기법과 바둑판식기법이다. 랜덤기법은 좁은 공간에서는 어느 정도 효과적으로 청소를 수행하지만, 넓은 공간에서는 청소가 올바르게 수행되지 않는다. 바둑판식기법은 조밀하게 청소를 진행하기 때문에 장애물이 없거나 작은 공간에서 작업 성능이 뛰어나다. 그러나 바둑판식기법으로 복잡한 공간을 청소할 때는 청



소를 완료하는 시간이 증가한다.

청소로봇이 청소를 효율적으로 진행하기 위해서는 작업공간을 탐색하여 청소작업을 계획적으로 진행해야 한다. 본 논문에서 제안한 가상지도를 이용한 경로계획기법은 청소의 효율성을 높이기 위해 가상지도를 이용하여 외곽선순회기법으로 청소를 수행한다.

랜덤기법을 이용한 청소로봇 R1은 시뮬레이션에서 제공한 모든 환경에서 청소를 완료하지 못하였다. 바둑판식기법을 이용한 청소로봇 R2는 작업공간이 단순한 구조에서는 높은 청소 성능을 나타냈지만, 작업공간의 복잡도가 증가할수록 매우 낮은 청소 성능을 나타내었다. 환경탐색을 통해 가상지도를 생성하고 바둑판식기법으로 청소를 수행하는 청소로봇 R3는 가상지도를 생성하지 않는 청소로봇 R2보다 높은 성능을 나타내었다. 이와 같은 시뮬레이션 결과로 환경탐색에 대한 중요성을 확인할 수 있었다. 가상지도와 외곽선순회기법을 사용한 R4는 R3보다 우수한 성능을 나타내었다. 가상지도 생성이 효율적인 청소를 수행하는 데 중요한 요소로 작용하며, 외곽선순회기법이 바둑판식기법에 비해 청소 효율성이 뛰어나다는 것을 본 시뮬레이션을 통해 확인할 수 있었다. 시뮬레이션에서 R4는 R2에 비해 평균 34.99% 성능 향상을 나타내었고, R3에 비해 평균 10.44%의 성능 향상을 나타내었다.

향후 연구로는 작업공간의 형태와 같은 지형구조의 다각적 분석을 통해 가상지도에 환경정보가 많이 표현되도록 하여 청소 성능을 향상시키는 연구가 필요하다.

## 참고문헌

- [1] 김병수, "가정용 로봇 산업 현황," 정보통신진흥연구원, 2004년 10월.
- [2] 신경철, "지능형 로보틱스 산업동향," 한국컴퓨터정보학회지, 제 14권, 제 1호, 11-32쪽, 2006년 6월.
- [3] J. L. Jones, "Robots at the Tipping Point: The Road to the iRobot Roomba," IEEE Robotics & Automation Magazine, Vol. 13, No. 1, pp. 76-78, Mar. 2006.
- [4] Y. Oh and Y. Watanabe, "Development of Small Robot for Home Floor Cleaning," In Proceedings of the 41st SICE Annual Conference, Vol. 5, pp. 3222-3223, Aug. 2002.
- [5] E. Prassler, A. Ritter, C. Schaeffer, and P. Fiorini, "A Short History of Cleaning Robots," Autonomous Robots, Vol. 9, No. 3, pp. 211-226, Dec. 2000.
- [6] P. Fiorini and E. Prassler, "Cleaning and Household Robots: A Technology Survey," Autonomous Robots, Vol. 9, No. 3, pp. 227-235, Dec. 2000.
- [7] J. Forlizzi and C. DiSalvo, "Service Robots in the Domestic Environment: A Study of the Roomba Vacuum in the Home," In Proceeding of the 1st ACM SIGCHI/SIGART Conference on Human-Robot Interaction, pp. 258-265, Mar. 2006.
- [8] R. N. Carvalho, H. A. Vidal, P. Vieira, and M. I. Ribeiro, "Complete Coverage Path Planning and Guidance for Cleaning Robots," In Proceedings of the IEEE International Symposium on Industrial Electronics, Vol. 2, pp. 677-682, Jul. 1997.
- [9] H. Choset, "Coverage of Known Spaces: The Boustrophedon Cellular Decomposition," Autonomous Robots, Vol. 9, No. 3, pp. 247-253, Dec. 2000.
- [10] G. Schmidt and C. Hofer, "An Advanced Planning and Navigation Approach for Autonomous Cleaning Robot Operations," In Proceedings of the IEEE International Conference on Intelligent Robots and System, Vol. 2, pp. 1230-1235, Oct. 1998.
- [11] S. C. Wong, L. Middleton, and B. A. MacDonald, "Performance Metrics for Robot Coverage Tasks," In Proceedings of Australasian Conference on Robotics and Automation, pp. 7-12, Nov. 2002.
- [12] M. T. Ribes, "Optimization of Floor Cleaning Coverage Performance of a Random Path-Planning Mobile Robot," Informatica, pp. 1-31, Jun. 2007.
- [13] I. Rekleitis, A. P. New, E. S. Rankin, and H. Choset, "Efficient Boustrophedon Multi-Robot Coverage: an algorithmic approach," In Annals of Mathematics and Artificial Intelligence, Vol. 52, No. 2, pp. 109-142, Apr. 2008.
- [14] 안상선, 신성욱, 이정욱, 백두권, "실내 환경에서 운영 가능한 RFID 기반 멀티 로봇 관리 시스템," 한국컴퓨터정보학회논문지, 제 13권, 제 6호, 2008년 11월.
- [15] P. Fiorini and E. Prassler, "Cleaning and Household Robots: a Technology Survey," Autonomous Robots, Vol. 9, No. 3, pp. 227-235, Dec. 2000.
- [16] J. S. Oh, J. B. Park, and Y. H. Choi, "Complete Coverage Navigation of Clean Robot based on Triangular Cell Map," In Proceedings of the IEEE International Symposium on Industrial Electronics, Vol. 3, pp.

2089-2093, Jun. 2001.

- [17] E. U. Acar, H. Choset, Y. Zhang, and M. Schervish, "Path Planning for Robotic Demining: Robust Sensor-based Coverage of Unstructured Environments and Probabilistic Methods," *International Journal of Robotics Research*, Vol. 22, No. 7-8, pp. 441-466, Jul. 2003.
- [18] Y. Gabriely and E. Rimon, "Spanning-tree based Coverage of Continuous Areas by a Mobile Robot," *In Annals of Mathematics and Artificial Intelligence*, Vol. 31, No. 1-4, pp. 77-98, May. 2001.
- [19] K. S. Jung, Y. J. No, Y. C. Lim, and Y. J. Ryoo, "Path-Planning for Cleaning Robot Using a Wall Tracing," *In Proceedings of the International Conference of Control, Automation and Systems*, pp. 2125-2128, Oct. 2002.
- [20] C. Hofner and G. Schmidt, "Path planning and guidance techniques for an autonomous mobile cleaning robots," *Robotics and Autonomous Systems*, Vol. 14, No. 2-3, pp. 199-212, May. 1995.

### 저자 소개



#### 김형일

1996 ~ 1998 : (주)경기은행

2004 :

동국대학교 컴퓨터공학과(공학박사)

2005 ~ 2006 :

동국대학교 컴퓨터공학과 IT교수(정보통신부)

2007 ~ 현재 :

나사렛대학교 멀티미디어학과 교수

관심분야 : 임베디드시스템, 지능형시스템, 의료정보, 추천시스템, 정보검색, 인공지능