

이동 ROBOT를 위한 경로계획법

범 희 락 조 형 석
한국과학기술원 생산공학과

Path Planning Method for Mobile Robot

Hee Rak Beom Hyung Suck Cho
Department of production Engineering
Korea Advanced Institute of Science and Technology

ABSTRACT

This paper proposes a new path planning method for obstacle avoidance of mobile robot. In order to achieve easy planning of the path, a simple representation of the empty space is achieved based on thinning algorithm. The proposed planning technique facilitates the direct use of information obtained by camera. Comparing to the V-graph method, the task of determining the shortest path from the resulting skeleton of empty space is optimized in terms of number of computation steps. The usefulness of the proposed method is ascertained by simulation.

게 찾아낼수 있다. 그러나 Quad Tree는 환경에 대한 대략적인 기술 방법이며 선택된 좌표계에 따라 다르므로 어떤 경우에는 환경에 통과 가능한 경로가 있을지라도 경로를 찾지 못하는 단점이있다.

그리하여 본 논문에서는 환경에 놓여 있는 장애물의 형태가 매우 복잡한 경우이거나, 로봇의 출발지점과 목표지점이 수시로 변화하는 경우에 로봇이 장애물과 충돌하지 않고 갈수있는 최단경로를 찾는 새로운 방법을 제시하고자 한다. 본 논문에서는 이동로봇의 경로계획을 위하여 환경(Environment)을 임의의 형상을 한 장애물(B)들로 구성된 2차원 공간으로 가정하고, 로봇(RO)은 2자유도를 가진 원형으로 가정한다. 또한 환경에 대한 정보로서, 카메라로 부터 얻은 영상에 간단한 필터링(Filtering)과 쓰레쉬홀딩(Thresholding)을 거친 이진 영상(Binary image)을 이용한다.

1. 서론

경로 계획은 이동 로봇에 지능을 부여하기 위한 것 중의 하나이다 [1]-[4]. 문제는 장애물(B)이 놓여있는 환경하에서 로봇(RO)이 출발지점(S)에서 목표지점(G)사이의 모든 위치에서 $RO \cap B = \emptyset$ 이 되는 경로 P를 어떻게 찾아 내느냐이다. 경로를 찾기위하여 먼저 로봇의 기하학적 또는 기구적인 표현 방법과 환경에 대한 기하학적 표현 그리고 출발위치와 목표 위치등이 필요하다. 경로를 찾는 방법으로는 크게 포텐셜(Potential)방법과 형상공간(Configuration space) 방법 두가지로 분류된다. 포텐셜 방법[4][5]은 포텐셜 함수의 최소화에 바탕을 두고 있으므로 국소적인 최소점(Local minima)에 빠지는 데드락(Dead lock) 현상이 발생하며 데드락 현상에 대한 일반적인 해는 존재하지 않는다는 단점이 있다. 형상공간 방법으로는 Lozano-Perez가 처음으로 서로 마주 보이는 꼭지점(Vertex)들로 연결된 그래프인 V-그래프를 이용하여 다각형 형태의 장애물이 놓여 있는 환경에서, 로봇을 하나의 점으로 가정하고 경로를 찾는 문제를 다루었다[1]. 그러나 V-그래프 방법은 환경이나 로봇의 크기에 따라 V-그래프가 변화하기 때문에 V-그래프는 다시 작성 되어야 한다는 단점이있다. 이러한 단점을 극복하기 위하여 H.Noborio등은 환경을 Quad Tree로서 모델링 하는 효율적인 알고리즘을 제안하였다[6]. Quad Tree는 환경을 계층적인 구조로서 자유영역(Free region), 장애물 영역 (Ostacle region) 그리고 장애물영역과 자유영역이 혼합되어있는 혼합영역(Intersection region)으로 표시되므로 로봇이 통과 할수있는 자유영역(Free space)을 쉽

2. 환경의 표시 방법

지금까지 이동 로봇을 위한 환경의 모델링에 대하여 많은 연구가 이루어져 왔으며 그중 몇가지는 문제의 특정 부분에 초점을 두어 일반적인 해를 찾는데만 몰두하고 있다는 느낌을 준다[3][6]. 이러한 방법은 경로 계획을 하는데는 쉬울지 모르지만 환경에 대한 모델링은 매우 복잡해지는 단점이 있다. 그리하여 R. Brooks [7]는 몇가지의 도형을 이용하여 환경을 표시하였다. 그러나 작업장 내의 환경이 매우 복잡하여 지도를 작성하기 어려운 경우는 카메라를 이용한 방법이 유용하다. 작업장에 부착되어 있는 카메라로부터 현재의 로봇 위치 뿐만 아니라 작업장의 상태를 알수 있으며 이것으로부터 로봇의 경로계획을 한다면 로봇이 장애물과 충돌없이, 현재의 위치에서 목표점까지 갈 수 있을 것이다. 그러나 비전 데이터로부터 모든 거리 정보를 알아내기는 시간이 많이 걸리기 때문에 카메라를 작업장 바닥에 대하여 수직으로 설치하는 경우에는 장애물의 크기등을 쉽게 알 수 있다. 본 논문에서는 카메라가 작업장 바닥에 대하여 수직으로 설치되어 있다고 가정하고, 카메라 영상에 나타난 장애물에 대하여 로봇의 크기를 고려한 영상을 가지고 경로 계획을 수행할 수 있다. 카메라로 부터 얻은 데이터에 로봇의 크기를 고려하기 위하여 장애물을 확장시키고 로봇을 하나의 점으로 축소시키는 Lozano Perez가 제시한 방법을 사용하였다. 이렇게 함으로써 이동 로봇을 하나의 점으로 볼 수가 있으며, 영상중의 한개의 화소(Pixel)가 로봇의 크기가 된다. 그림 1은 작업장의 장애물에 대하여 로봇의 크기를 고려한 환경에 관한 데

이터이다. 그림 1과 같은 데이터로부터 로봇이 안전하게 지나갈 수 있는 영역을 구하기 위하여 Thinning 알고리즘을이용하였다.

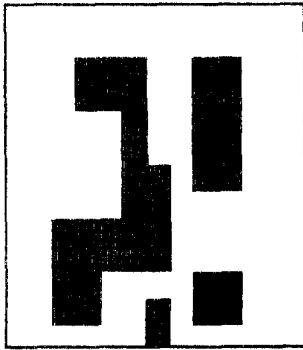


그림1 환경을 나타내는 이진영상.

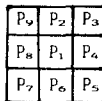
Thinning 알고리즘 [8]

빈 공간을 하나의 물체로 가정하여 Thinning 알고리즘을 적용하면 그 물체의 골격만 남게 된다. 그림 2에서 보는 바와 같이 알고리즘의 첫째 단계에서 물체를 구성하고 있는 화소들이 원호를 구성하고 있지 않거나 또는 그화소를 제거하더라도 8-Connectivity를 만족하면 'L'로 표시된 물체의 왼쪽의 경계화소들이 제거된다. 두번째 단계에서 왼쪽 경계화소들에 대하여 했던 똑같은 방법으로 'R'로 표시된 화소들을 제거한다. 그런후에 위쪽(T)과 아래쪽(B)의 경계면의 화소들에 대해서도 위와 같은 절차를 적용시킨다. 이러한 일련의과정을 거치게 되면 더 이상의 화소들이 제거되지 않게되며 결과적으로 물체의 골격만 남게 된다.

Thinning 알고리즘은 다음과 같다.

제 1단계 : 아래의 조건이 만족되면 P_1 을 제거한다.

- a) $2 \leq N(P_1) \leq 6$
- b) $S(P_1) = 1$
- c) $P_2 \cdot P_4 \cdot P_6 = 0$
- d) $P_4 \cdot P_6 \cdot P_8 = 0$



제 2단계 : 아래의 조건이 만족되면 P_1 을 제거한다.

- a) $2 \leq N(P_1) \leq 6$
- b) $S(P_1) = 1$
- c) $P_2 \cdot P_4 \cdot P_6 = 0$
- d) $P_2 \cdot P_4 \cdot P_6 = 0$

여기서, $N(P_1)$: P_1 의 이웃한 화소 들 중에서 0이 아닌 화소들의 수
 $S(P_1)$: P_1, P_2, \dots, P_9 에서 0-1의 변화가 발생한 회수

Thinning에 의하여 생긴 빈 공간의 골격에는 매우 중요한 정보가 들어 있다. 빈 공간의 골격을 이루고 있는 화소들의 좌표가 로봇이 지나갈 수 있는 경로점들이 되며 이러한 좌표점들의 집합이 하나의 안전한 경로를 이루게된다. 그러나 Thinning 과정은 Thinning의 순서에 따라 골격의 모양이 조금씩 변화하게 되며 8-Connectivity 조건을 만족시키지 못하는 단

점이 있다. 따라서 본 논문에서는 적절한 오퍼레이터를 이용하여 8-Connectivity를 만족하도록 하였으며 8-Connectivity 가 경로들의 Node 를 발견하는데 중요한 구실을 한다. 다시말해 통과가능한 경로의 수를 결정하거나 각 Node사이의 Reachability를 찾는데 사용되고 있다.

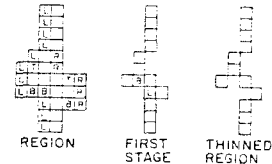


그림2 Thinning 알고리즘의 예.

3. 경로 계획

로봇이 가장 안전하게 통과할 수 있는 경로는 그림 3과 같이 Thinning에 의하여 얻은 빈 공간의 골격이 되며 S점에서 G점으로 가는 실제의 경로점들은 영상 데이터에 Inverse Perspective Transformation을 함으로써 얻어질 수 있다. 영상 데이터의 크기를 $n \times n$ 이라고 할 때 실제의 공간 $M = \{(x_i, y_j) \in R^2, 0 \leq i, j \leq (n-1)\}$ 으로 구성된 좌표점들의 집합으로 볼 수 있다. 여기서 x_i, y_j 는 영상 데이터에 Inverse Perspective Transformation 함으로써 얻어진다. 또한 실제 공간은 장애물 영역 (B) 과 통과 가능한 영역 (Q)으로 구성되며 현재 로봇의 위치를 RO라 하면 $RO = \{(x_r, y_r) \in R^2\}$ 로 표시된다. S지점에서 G지점까지의 충돌회피 경로 계획을 한다는 것은 $B \cap RO = \emptyset$ 인 공간의 좌표점 (x_i, y_j) 의 시퀀스(Sequence)를 찾는 문제로 되며 통과 가능한 영역 Q는 $Q = U\{(x_i, y_j) \in R^2; (x_i, y_j) \in B \cap RO, i, j < n\}$ 이다. 또한 $RO, B \subseteq M$ 에 대하여 Euclidean 거리 d를 다음과 같이 정의한다.

$$d(RO, B) = \inf\{(x_r, y_r), (x_0, y_0) : (x_r, y_r) \in RO, (x_0, y_0) \in B\}$$

여기서 $d(RO, B)$ 가 클수록 장애물과의 충돌 없이 통과할 수 있으며 $d(B, RO)$ 이 최대가 되는 좌표점이 가장 안전한 경로가 되고 그경로가 빈 공간의 골격을 이루고 있는 좌표점들이 된다.

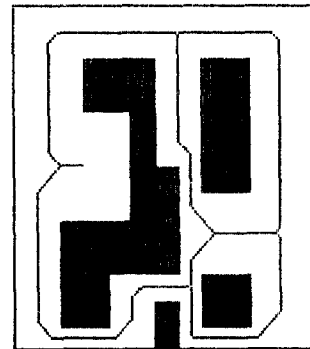


그림3 빈 공간의 골격.

Reachability 그래프

본 논문에서의 Node는 다음과 같이 정의 할수있다.

Node 정의:

빈 공간에 대하여 Thinning 알고리즘을 적용하여 얻은 골격은 8-Connectivity 조건을 만족함으로 3x3 윈도우(Window)내에 0이 아닌 최소의 수가 4일 때 윈도우의 중앙에 위치하는 최소가 Node가 된다.

로봇이 S점에서 G점으로 가는 경로의 수는 그림 4에서 볼 수 있듯이 경로들이 만나는 점(Node)들의 수에 좌우된다. 먼저 경로를 구하기 위하여 Node를 구해야 한다. 3x3 윈도우를 전체 화면에 이동시켜 가면서 위의 조건이 만족될 때의 윈도우의 중앙 최소가 Node이며 Node의 좌표는 그때의 최소의 좌표가 된다. 이러한 일련의 과정을 거치게 되면 화면 내의 모든 Node들을 구할 수 있으며 이 Node들은 Reachability 그래프를 구성하는데 중요한 요소가 된다. 이렇게 하여 구해진 Node들로부터 Reachability 그래프를 구성하기 위하여 Node A₁에서부터 Thinning된 빈 공간의 골격을 따라 윈도우를 이동시킨다. 그리하여 새로운 Node A₂가 나타나면 Node A₁과 A₂는 서로 Reachable하며 Node A₁과 A₂를 연결하는 링크(Link)를 만든다. 그때의 링크는 A₁과 A₂사이의 좌표점으로 이루어진 Ordered Sequence가 된다. 이러한 기억된 좌표들이 로봇이 통과할 수 있는 중간 경로점들이 된다. 이렇게하여 구성된 그래프가 그림 4이고 이것을 Reachability 그래프라 한다. Reachability 그래프는 Node들 사이의 연결된 정보만 가지고 있으므로 그림 4에서 보는 바와 같이 출발 지점에서 목표지점 G까지 갈 수 있는 경로는 매우 많이 존재 하게 된다. 따라서 로봇이 한번 지나갔던 Node는 다시 지나가지 않도록 트리(Tree)구조의 방향성을 지니게 한 그래프가 그림 5이다. 그림 5에서 통과 가능한 경로들 중에서 안전하고 빨리 목표점에 도달할 수 있는 경로를 구하는 것이 다음의 목표가 되었다.

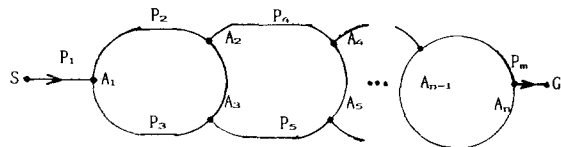


그림 4 Reachability 그래프.

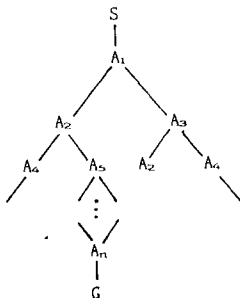


그림 5 Tree 구조의 Reachability 그래프.

최단 경로

최단 경로를 구하기 위하여 먼저 Cost function을 정의 해야한다. 본 논문에서는 장애물을 로봇의 크기만큼 확장하고 로봇을 하나의 점으로

축소시켰기 때문에 될수 있으면 장애물의 꼭지점이나 장애물의 면을 따라 진행하면 최단 거리로 목표점에 도달할 수 있다. 로봇의 이동경로 P는 R² Space에서의 좌표점들로 이루어져 있기때문에출발점(S)에서 목표지점(G)까지 갈수있는 경로(P_{SG})가 h개의 좌표점들로 이루어져 있다고 하면 P_{SG} = {(x_k, y_k) ∈ R², k=1, 2, ..., h} ∪ {(x_k, y_k) ∈ R²; k=1, 2, ..., h} 로 표시할수 있다. 현재 로봇의 위치(R0)를 X₁라 하고 서로 인접한 경로점의 좌표점을 X_i = (x_i, y_i), X_{i+1} = (x_{i+1}, y_{i+1})라 하면 두점 사이의 거리(L_i)는 L_i = ((x_{i+1} - x_i)² + (y_{i+1} - y_i)²)^{1/2} 이 된다. 따라서 Cost function은 다음과 같이 정의 된다.

$$J = \sum_{i=1}^{h-1} (L_i) + \epsilon \cdot d(R0, B)$$

여기서 ε은 안전도를 나타내는 요소이고 목표점 까지 가는 경로의 길이와 로봇과 장애물과의 거리중에서 어디에 비중을 둘것인가에 따라 달라진다. 본 논문에서는 ε을 0으로 하여 최단 경로를 구하였다. 그림 6에서 볼수있듯이 로봇의 경로가 되는 빈공간의 골격들은 장애물과 떨어져 있다. 문제는 로봇이 장애물과의 충돌을 회피하면서 목표지점에 빨리 도달할 수 있도록 하기위하여 경로를 구성하고 있는 좌표점들을 어떻게 재 배치하느냐이다.

그림 6에서 우리가 경로 S A₀ B₀ C₀ D₀ E₀ F₀ H₀ I₀ G점들을 알고있지못하다면 로봇의 방향을 목표점으로하여 이동시키면서 장애물과 만나는 점을 구하고 장애물의 면을 따라 이동하다가 장애물이 없으면 다시 방향을 목표지점으로 하여 이동하면 결국 목표지점에 이르게 된다. 이러한 방법은 국소적인의미의 경로 계획 이 되며 전체적인 의미에서의 경로 계획은 아니다. 따라서 그림6의 경로점들은 Thinning에 의하여 이미 알고있는 좌표점들이므로 이것을 이용하여 좌표점 들을 하나씩 줄여 나간다. 먼저 출발지점 S에서 A₀지점까지 직선으로 연결한다음 그직선상에서 장애물의 존재 유무를 체크한다. 장애물이 존재하지 않은 경우에는 직선상에 있는 점들이 곧 새로운 경로점들이 되며 장애물이 존재하는 경우에는 경로점들은 바뀌지 않는다. 이러한 과정을 거쳐 S지점과 H지점을 연결 했을 때에 연결된 직선 상에 장애물이 없으므로 경로점들은 S A₁ B₁ C₁ D₁ E₁ F₁ H₀ I₀ G으로 변한다. 또 S와 I₀를 연결 하였을 때에 직선상에 장애물이 존재하므로 경로점들은 변하지 않게 된다. 이러한 방법으로 G점까지 수행한후에 다시 출발 지점을 A₁으로 하여 G지점까지 반복한다. 이러한 작업을 반복적으로 하게 되면 더이상 Cost funtion은 변화하지 않게 되며 이때의 경로가 Local Minimum이 되며 Reachability 그래프에 나타난 모든 경로에 대하여 위의 방법을 수행하게 되면 Global Minimum이 되는 경로를 구할 수 있다.

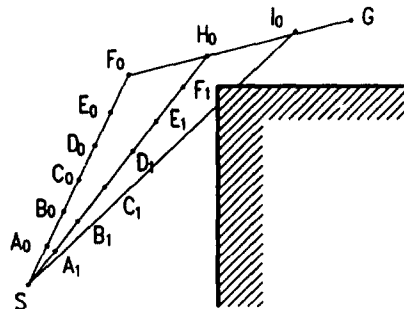


그림 6 최단 경로를 찾는 방법.

4. 시뮬레이션 결과

위와 같은 방법이 타당한가를 조사하기 위하여 시뮬레이션을 해보았다. 시뮬레이션에 사용된 환경에 대한 데이터는 그림 1과 같이 4개의 장애물로 구성되어 있다. 먼저 환경에 대하여 Thinning 알고리즘을 적용시키면 그림 3과 같이 되며 그림3에서 출발지점 S와 목표지점 G를 입력하면 그림 7과 같은 경로들이 형성된다. 그림 8은 위의 방법을 사용해 경로를 줄여 나가는 과정을 보여 주고 있다. 그림 9는 모든 경로들중에서 Cost function 이 최소가 되는 경로이다.

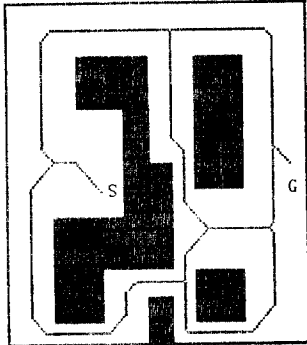


그림7 S점과 G점사이의 모든 경로.

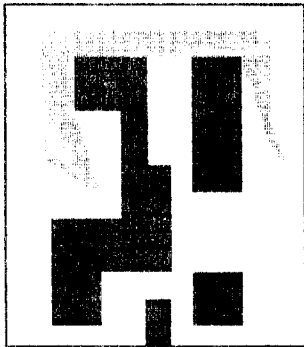


그림8 직단경로 찾는 과정.

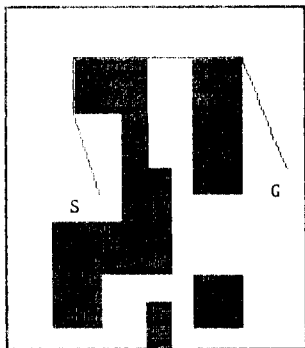


그림9 최단경로.

5. 결론

본 논문에서 빈공간을 Thinning 알고리즘을 이용하여 모델링하고, 모델링된 빈공간의 골격을 사용하여 최단경로를 구해본 결과 V-그래프를 이용하여 얻은 결과와 유사하다. 그러나 Thinning에 의하여 Reachability 그래프를 작성하게 되면 다음과 같은 장점이 있다. 경로의 분기점이 되는 Node의 수는 장애물의 갯수에 의해 결정이 되며, 장애물의 형태에 크게 영향을 받지 않으므로 복잡한 장애물이 놓여 있는 환경에 대하여 효율적이다.

참고문헌

- [1] Tomas Lozano-Perez and M.A.Wesley, "Algorithm for planning collision-free paths among polyhedral obstacles", Commun. ACM, Vol. 22, NO. 10, pp 560-570, OCT. 1979.
- [2] Tomas Lozano-Perez, "Automatic planning of manipulator transfer movements", IEEE Trans. Syst., Man, Cybern., Vol. SMC-11, pp 681-698, 1981.
- [3] Tomas Lozano -Perez, " Spatial planning : a configuration space approach", IEEE Trans. on Computer C-32, No.2, pp 108-120, Feb. 1983.
- [4] O. Khatib, " Real-time obstacle avoidance for manipulators and mobile robots", Proc. IEEE Int. Conf. Robotics and Automation, pp 500 -505, St. Louis, March 1985.
- [5] Charles W. Warren, "Global path planning using artificial potential fields", Proc. IEEE Int. Conference. on Robotics and Automation, pp316-321, 1989.
- [6] H. Noborio, T. Naniwa and S. Arimoto, " A fast path planning algorithm by synchronizing modification and search of its path - graph ", Proc. IEEE international Workshop on Artificial Intelligent for Industrial Application, pp 351-357, 1988.
- [7] R. Brooks, "Solving the find path problems by good representation of free space", IEEE Trans. on Syst., Man, Cybern. Vol., SMC-13, N03 March/April PP190-197, 1983.
- [8] Pratt, "Digital image processing", John Wiley, pp517-520, 1984.