

원형작업공간의 기하투영에 의한 일차 매개 곡선을 이용한 충돌회피 궤적 계획

A Collision-Free Path Planning Using Linear Parametric Curve Based on Circular Workspace Geometry Mapping

°남궁 인

한국원자력연구소(Tel: 042-868-8774, E-mail: inamgung@nanum.kaeri.re.kr)

Abstract A new algorithm for planning a collision-free path is developed based on linear parametric curve. A collision-free path is viewed as a connected space curve in which the path consists of two straight curve connecting start to target point. A single intermediate connection point is considered in this paper and is used to manipulate the shape of path by organizing the control point in polar coordinate (θ, ρ) . The algorithm checks interference with obstacles, defined as GM (Geometry Mapping), and maps obstacles in Euclidean Space into images in CPS (Connection Point Space). The GM for all obstacles produces overlapping images of obstacle in CPS. The clear area of CPS that is not occupied by obstacle images represents collision free paths in Euclidian Space. Any points from the clear area of CPS is a candidate for a collision-free path. A simulation of GM for number of cases are carried out and results are presented including mapped images of GM and performances of algorithm.

Keyword Robot Path Planning, Geometry Mapping, Obstacle Avoidance

1. 서론

로봇의 장애물회피 및 궤적계획은 상위레벨/작업레벨에서의 로봇 제어를 위한 중요한 요소중의 하나이다. 장애물회피와 궤적계획은 로봇프로그래밍의 많은 부분을 단순화시켜서 단지 출발점과 종착점의 위치와 자세 등을 지정해주면 로봇이 지나야할 궤적을 구하고, 장애물을 회피하는 작업을 가능하게 한다. 이러한 로봇의 장애물회피 및 궤적계획은 로봇프로그래밍의 능력을 증대시키고, 작업장내에서 충돌등 사고를 줄여주고, 작업자와 컴퓨터 간의 연계를 증진시켜준다.

일반적으로 로봇 궤적계획은 주어진 두점 사이를 연결하는 장애물과 교차하지 않는 궤적을 구하는 문제이다. 이러한 로봇 궤적계획의 알고리즘으로 visibility graph search[1], free space search[2], potential field method[3], bug's algorithm[4] 등이 연구되었다. Visibility graph search는 장애물의 각 꼭지점과 출발점, 종착점사이를 직선으로 연결한다. 이 연결선 중에서 장애물과 교차하지 않는 연결선으로 구성된 출발점으로부터 종착점까지를 트리구조의 연결선으로 구성한다. 이 트리구조의 연결선으로부터 최단거리 궤적을 찾는다. Free space search는 작업공간을 여러 개의 셀로 나누고, 각셀 내의 장애물 존재여부로 셀을 빈셀(empty cell), 혼합셀(mixed cell) 및 충만셀(full cell)로 구분한다. 혼합셀은 더작은 셀로 분할하여 장애물의 존재여부를 판단하여 셀을 구분한다. 각 단계마다 빈셀을 트리구조로 구성하여 출발점과 종착점을 연결하는 궤적을 찾는다. Potential field 방법에서는 장애물을 척력(repulsive pole)으로 나타내고, 종착점을 인력(attractive pole)으로 나타낸다. 장애물회피 궤적은 출발점과 종착점을 연결하는 궤적으로 optimization search method를 사용하여 구한다. Bug's 알고리즘은 벌레의 움직임을 본따서 출발점으로부터 종착점까지를 직선과 장애물의 외면으로 연결한다. 출발점으로부터 종착점으로 움직여가며, 장애물을 만나면 장애물의 벽을 따라 종착점으로 움직인다. 그래서 다시 종착점으로 연결된 직선을 만나면 그 궤적을 따라 종착점까지 움직여간다.

장애물 회피 궤적은 출발점과 종착점을 연결하는 직선 선분의 조합이거나, 직선과 곡선의 조합 또는 곡선의 조합으로 볼 수 있다. 그림 1 참조. 출발점, S,과 종착점, T,을 연결하는 직선은 점선으로 나타낸 것인데 장애물과 간섭을 한다. 장애물회피 궤적은 연결된 직선으로 나타낸 것이다. 여기서 중간연결점은 궤적이 장애물과 간섭하지 않도록 궤적의 모양을 제어하는데 사용된다.

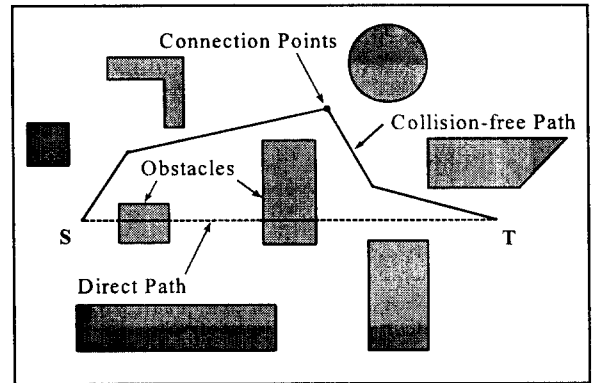


그림 1. 연결된 선분으로 이루어진 충돌회피궤적

Figure 1. connected line segments for collision-free path
본 논문에서는 일차 매개곡선을 사용하고 하나의 중간 연결점이 있는 경우의 궤적 계획문제를 다루었다. 최종궤적이 하나의 중간연결점만을 가지고 있다고 가정함으로써, 문제를 단순화 시켰다. 여기서 중간연결점의 위치를 극좌표로, (θ, ρ) , 나타내어 궤적과 장애물의 간섭여부를 중간연결점 공간으로 (CPS, Connection Point Space) 투영하였고 기하매핑(Geometry Mapping)으로 정의하였다. 이렇게 정의된 원형작업공간이 (CWS, Circular Workspace) 실제작업공간을 (AWS, Actual Workspace) 전부 포함해야한다. 그림 2 참조.

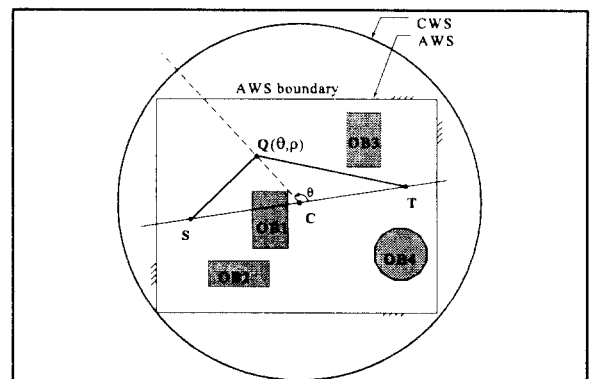


그림 2. 극좌표로 나타낸 중간연결점과 원형작업공간
Figure 2. Polar coord. of connection point and CWS

2. 일차 매개곡선을 이용한 궤적계획

매개변수를 이용한 곡선은 다중 값을 갖는 함수 및 미분 값이 무한으로 발산하는 경우에도 수학적 어려움 없이 표현할 수 있고 방향성을 가지고 있는 여러 가지 이점을 가지고 있다 [5]. 그래서 그림 2의 직선 궤적 ST를 매개변수, s ,를 이용하여 다음과 같이 표현한다.

$$r(s) = (T - S)s + S \quad (1)$$

변수 s 를 0에서 1로 변화시켜가면 $r(s)$ 로 표현되는 궤적은 출발점, S , $r(s)_0$,에서 종착점, T , $r(s)_1$,으로 움직여간다. 따라서 매개변수를 이용한 궤적은 바로 로봇의 제어에 적용될 수 있다. 또한 매개곡선의 이러한 방향성은 두직선간의 간섭을 간단히 표현할 수 있게 한다. 여기서 선분 ST와 장애물의 모서리 P_1, P_2 간의 교차여부는 다음의 식을 이용하여 구한다.

$$(T - S)s + S = (P_j - P_i)t + P_i \quad (2)$$

여기서 s 는 선분 ST를 나타내는 매개변수이고, t 는 장애물의 모서리를 나타내는 매개변수이다. (2)로부터 s 와 t 에 대한 식을 얻을 수 있다.

$$s = \frac{(P_{ix} - S_x)(P_{iy} - P_{iy}) - (P_{iy} - S_y)(P_{ix} - P_{ix})}{(T_x - S_x)(P_{iy} - P_{iy}) - (T_y - S_y)(P_{ix} - P_{ix})} \quad (3)$$

$$t = \frac{(T_x - S_x)(S_y - P_{iy}) - (T_y - S_y)(S_x - P_{ix})}{(T_x - S_x)(P_{iy} - P_{iy}) - (T_y - S_y)(P_{ix} - P_{ix})} \quad (4)$$

만일 s 와 t 가 모두 $0 \leq s \leq 1$ 과 $0 \leq t \leq 1$ 의 범위 안에 들 경우, 선분 ST는 모서리 P_1, P_2 와 교차하게되고, 그렇지 않은 경우 교차는 일어나지 않는다. 이식에서 분모가 0이 될 경우, s 와 t 는 무한대로 발산하게되는데, 이것은 ST와 P_1, P_2 가 평행할 경우 발생한다. 따라서 식 (3)과 (4)의 계산은 분자와 분모를 비교하여, 만일 분자가 클 경우 선분 ST와 P_1, P_2 의 교차가 일어나지 않으므로 s 와 t 의 계산은 불필요하다.

작업공간내의 장애물은 3가지로 구분할 수 있는데, 하나는 장애물이 선 ST와 교차하는 경우이고, 두번째는 선 ST의 상부에 존재하는 경우, 그리고 마지막으로 선 ST의 하부에 존재하는 경우로 구분할 수 있다. 따라서 장애물과 선 ST간의 간섭이 없을 경우, 그림2의 OB2에 해당, 장애물의 위치를 결정하기 위한 추가적인 계산이 필요하다. 이 계산은 식(6)을 사용하여 삼각함수를 이용하지 않고 위치를 결정할 수 있다.

$$A = |STP_i| \quad (5)$$

실제로 (5)는 삼각형 STP_i의 면적의 두배를 나타내고, (5)의 부호는 S, T, P_i의 순서에 따라 변한다. 따라서 그림2의 경우, A가 양일 경우 P_i는 선 ST의 위에 위치하고, 음일 경우 아래에 위치하며, 0일 경우 선 ST상에 위치한다.

3. 장애물과 궤적의 간섭계산

앞장에서는 출발점과 종착점을 연결하는 직선궤적과 장애물과의 교차에 대해 다루었다. 만일 그 궤적이 장애물과 간섭을 일으킬 경우 하나의 제어점, Q,을 중간 연결점으로 하여 그림2와 같이 궤적을 구성한다. 여기서 연결점을 극좌표(θ, ρ)로 나타내고, 매개변수의 정의 역은 $0^\circ < \theta < 360^\circ$ ($\theta = 0^\circ, 180^\circ, 360^\circ$)와 $0 < \rho \leq \rho_{max}$ 로 정의하여 유클리드공간에서의 실제작업공간 전체를 포함하도록 원형작업공간을 정의한다. θ 가 $0^\circ, 180^\circ, 360^\circ$ 일 경우 궤적은 ST와 일치하므로 의미가 없어 제외한다. 이 원형작업공간의 중심, C,은 선분ST의 중심점으로 정의한다. 원형작업공간을 구하기위해서는 실제작업공간을 구성하는 외곽점과 원형작업공간의 중심점으로부터 실제작업공간의 최대 내접반경, R_{max} ,을 구하여 원형작업공간의 반지름으로 한다. 이렇게 구한 R_{max} 로부터 θ 값에 따른 Q_{max} ($Q_{max,x}, Q_{max,y}$)을 다음과 같이 정의한다.

$$Q_{max,x} = \frac{R_{max}}{D} \{ (T_x - S_x)\cos\theta - (T_y - S_y)\sin\theta \} + C_x \quad (6)$$

$$Q_{max,y} = \frac{R_{max}}{D} \{ (T_x - S_x)\sin\theta + (T_y - S_y)\cos\theta \} + C_y$$

여기서 D는 선분 ST의 길이, $D = |T - S|$,이다. 이것을 사용하여 최외곽을 지나는 연결점의 위치를 구할 수 있다.

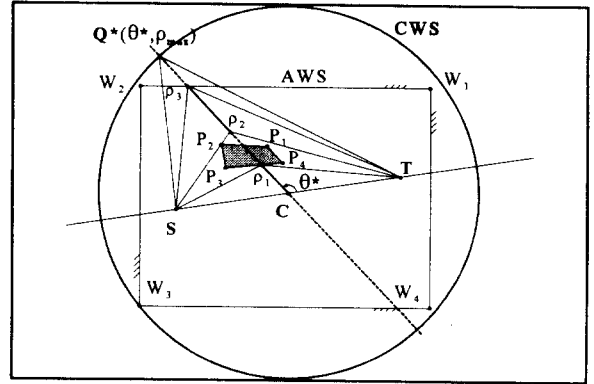


그림 3. 장애물과 궤적 사이의 간섭

Figure 3. Interference between obstacle and path

주어진 각도에 따라 최외곽 연결점의 위치를 (6)을 이용하여 구하고, 장애물과 간섭을 일으키는 변수 ρ 의 값을 구한다. 장애물과의 간섭은 선분 SQ, QT와 장애물모서리 사이의 교차여부를 판별하여 결정한다. 그림 3에 나타난 바와 같이 장애물과의 간섭은, i) 궤적이 장애물의 모서리를 지나는 경우와, ii) 연결점이 장애물의 모서리와 접촉하는 경우로 구분된다.

첫 번째 경우는 (7), (8)의 관계식으로 나타낸다.

$$\{ (Q_{max} - C)\rho + C - S \} r + S = P \quad (7)$$

$$\{ (Q_{max} - C)\rho + C - T \} s + T = P \quad (8)$$

위 식에서, 변수 r 과 s 는 각각 선분 SQ와 QT를 나타내고, 변수 ρ 는 연결점 Q의 위치를 나타내고, 점 P는 장애물의 모서리 점을 나타낸다. 위 식에는 변수 ρ 는 변수 r 과 s 를 소거함으로써 다음과 같이 유도할수 있다.

$$\rho = \frac{(P_x - S_x)(C_y - S_y) - (P_y - S_y)(C_x - S_x)}{(Q_{max,x} - C_x)(P_y - S_y) - (Q_{max,y} - C_y)(P_x - S_x)} \quad (9)$$

$$\rho = \frac{(P_x - T_x)(C_y - T_y) - (P_y - T_y)(C_x - T_x)}{(Q_{max,x} - C_x)(P_y - T_y) - (Q_{max,y} - C_y)(P_x - T_x)} \quad (10)$$

(9)는 선 SQ 상에서 Q의 위치를 나타내는 ρ 를 계산하고, (10)은 선 QT상에서 점 Q의 위치를 구한다. 이 두 ρ 값은 점 P가 CQ_{max} 선상에 있지 않는 한 서로 다르다. 만일 선 SP와 CQ_{max} 또는 TP와 CQ_{max} 가 서로 평행하면 (9)또는 (10)의 분모는 0이 된다. 이러한 조건은 ρ 값이 0과 ρ_{max} 사이인 점을 이용하여, 분자와 분모를 바로 비교함으로써 해결할 수 있다. 만일 두 ρ 값이 모두 $0 \leq \rho \leq \rho_{max}$ 일 경우, 점 P는 삼각형 $SQ_{max}T$ 의 내부에 존재하고, 더큰 ρ 값을 가지는 선분이 점 P와 교차하게된다. 그림 3 참조.

두 번째의 경우, 연결점이 장애물의 모서리와 접촉하는 경우, 선분 CQ_{max} 는 모서리와 교차하게된다. 이 경우 ρ 는 (2)를 이용하여 다음의과 같이 나타낼 수 있다.

$$(Q_{max} - C)\rho + C = (P_2 - P_1)t + P_1 \quad (11)$$

(11)에서 변수 ρ 와 t 는 각각 선분 CQ_{max} 와 모서리 P_1P_2 를 나타낸다. (11)으로부터 변수 ρ 와 t 는 다음과 같이 (12)와 (13)으로 나타낼 수 있다.

$$\rho = \frac{(P_{1,x} - C_x)(P_{2,y} - P_{1,y}) - (P_{1,y} - S_y)(P_{2,x} - P_{1,x})}{(Q_{max,x} - C_x)(P_{2,y} - P_{1,y}) - (Q_{max,y} - C_y)(P_{2,x} - P_{1,x})} \quad (12)$$

$$t = \frac{(Q_{max,x} - C_x)(C_y - P_{1,y}) - (Q_{max,y} - C_y)(C_x - P_{1,x})}{(Q_{max,x} - C_x)(P_{2,y} - P_{1,y}) - (Q_{max,y} - C_y)(P_{2,x} - P_{1,x})} \quad (13)$$

이식에서 선분 CQ_{max} 와 모서리 P_1P_2 가 서로 평행할 때 분모는 0

이 되고 이것은 첫 번째 경우와 같이 해결한다. 만일 ρ 와 t 가 둘 다 $0 < \rho < \rho_{max}$, $0 < t < 1$ 이면 두 선분간의 교차가 일어난 것이다.

이상과 같이 두 가지 경우의 계산을 적용하여, 궤적 SQ, QT와 장애물과의 간섭을 판별할 수 있다.

4. 원형작업공간의 장애물분류

앞에서 취급한 기하매핑은 주어진 각도(θ)에 따른 장애물과의 간섭여부를 변수 ρ 의 값으로 계산하는 것이었다. 본절에는 이러한 기본 작업에 바탕한 장애물의 이미지를 구축하는데 필요한 장애물의 분류와 기하매핑의 흐름에 대해 기술한다. 장애물의 분류는 장애물의 위치에 따라 간섭이 일어나는 ρ 의 영역을 판별하기 위한 것이다. 실제작업공간은 출발점과 종착점을 포함하고 있으므로 기타 장애물과는 별도로 간주하여 취급해야한다. 그림 4에 장애물의 분류에 대해 나타내었고 장애물과의 간섭구간을 나타내는 ρ 를 표시하였다. 표 1에는 장애물의 분류에 따른 조건과 적용공식, 간섭구간에 대한 것을 나타내었다.

표 1. 장애물의 위치에 따른 분류 및 간섭구간 ρ

Table 1. Obstacle location and interference range of ρ

	조건	간섭구간	관련식	관련 그림
OB1	outside SQT	none	5	(a)
OB2	intersect segment ST	$0 \leq \rho \leq \rho_1$	3,4,5,9,10	(a)
OB3	intersect line ST, int. segment CQ _{max}	$\rho_2 \leq \rho \leq \rho_{max}$	3,4,5,9,10, 12,13	(a)
OB4	int. line ST, intersect SQ _{max} , TQ _{max}	$\rho_3 \leq \rho \leq \rho_{max}$	3,4,5,12,13	(a)
OB5	within SQ _{max} T (ρ_1 edge contact)	$\rho_1 \leq \rho \leq \rho_2$	3,4,5,9,10, 12,13	(b)
OB6	within SQ _{max} T (ρ_1 vertex contact)	$\rho_3 \leq \rho \leq \rho_1$	3,4,5,9,10, 12,13	(b)
OB7	above ST, int. segment CQ _{max} , TQ _{max} or SQ _{max}	$\rho_1 \leq \rho \leq 1$	3,4,5,9,10, 12,13	(c)
OB8	above ST, int. segment TQ _{max} or SQ _{max}	$\rho_2 \leq \rho \leq 1$	3,4,5,9,10, 12,13	(c)
AWS	AWS edge contact, AWS vertex contact.	$\rho_{AWS} \leq \rho \leq 1$	9,10,12,13	(c)

장애물의 기하매핑은 장애물의 분류를 표1에 따라 수행하여, 간섭구간에 대한 정보를 구한 후, 그 간섭구간의 하한, ρ_{min} ,과 상한, ρ_{max} 에 대한 계산은 그에 따른 적용 식을 사용하여 구한다. 따라서 각각의 장애물에 대해 각도, θ ,를 0°에서 360°로 증가시키며 간섭구간을 나타내는 변수 ρ 를 연결점공간에 매핑함으로써 장애물의 이미지를 구한다. 이렇게 구한 장애물의 이미지는 서로 겹치는 이미지를 형성한다. 여기서 장애물의 이미지가 투영되지 않은 빈 공간이 나타나는데, 이 공간은 장애물과 간섭이 없는 모든 궤적을 나타낸다. 그림 5 참조.

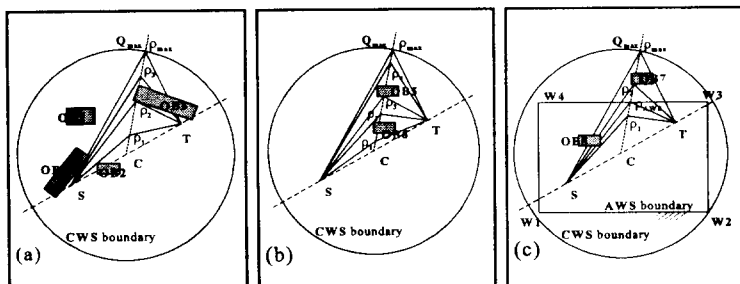


그림 4. 장애물의 위치에 따른 분류 및 간섭구간 ρ

Figure 4. Classification of obstacles and interference range ρ

5. 원형작업공간의 기하매핑 시뮬레이션

원형작업공간의 장애물에 대한 기하매핑에 있어서 장애물과 실제작업공간이 convex라는 가정을 하였다. 이것은 기하매핑의 가정이 아니고 시뮬레이션을 위한 가정이고, concave 장애물은 몇 개의 convex 장애물로 나뉘어 처리할 수 있다. 그림 5에서 OB2와 OB3는 이러한 분리된 장애물을 보여준다. 장애물의 위치에 따라 그 이미지는 몇 개의 특이한 패턴을 나타낸다. 기하매핑의 예로써, 그림 5의 윗부분은 작업공간을 나타내고, 아랫부분은 장애물의 이미지를 나타낸다. 여기서, 실제작업공간의, AWS, 외곽 모서리는 연결점공간의 상부를 점유하고, 각 모서리는 곡선으로 대응된다. 그리고 선분 ST와 교차하는 장애물은, OB1, 연결점공간의 하부 및 각도가 0°, 180° 및 360°인 부근을 차지한다. OB2와 OB3는 합쳐서 하나의 concave 장애물을 형성하고 각각 말굽형태의 이미지를 형성한다. OB5는 ST의 상부에 위치하고, 말굽형태의 이미지를 90°근처에 형성한다. OB4의 이미지는 두 부분으로 나뉘어 지는데, 실제로는 이미지가 360°를 주기로 반복되므로 하나이다. OB6는 선 ST를 S점 좌측에서 교차하므로 그 이미지는 180°근처에 형성된다. 이러한 시뮬레이션을 여러 가지의 경우에 대해 수행하여 그 결과를 표 2에 정리하였다. 각 경우에 해당하는 시뮬레이션의 플롯은 그림 5 (a) - (f)에 보여진다.

표 2. 기하매핑 시뮬레이션결과

Table 2. Result of Geometry Mapping simulation

	실행시간 (초)	설정환경	각도간격	성공여부
case 1	0.0736	(a)	3°	성공
case 2	0.0489	(b)	3°	성공
case 3	0.0599	(c)	3°	성공
case 4	0.0703	(d)	3°	성공
case 5	0.0665	(e)	3°	성공
case 6	0.1044	(f)	3°	실패

시뮬레이션 프로그램은 Boland C++/4.0을 사용하여 작성하였고, DOS/6.2 real mode와 윈도우95상에서 수행되었다. 윈도우95에서의 시뮬레이션은 윈도우의 특성상 일관적인 결과를 얻을 수 없었고, DOS에 비해 약 두배정도의 수행 시간이 요구되었다. Pentium/90MHz와 16MB RAM을 장착한 PC에서 DOS real mode에서의 시뮬레이션에 대한 결과가 표 2에 정리되어 있다. 장애물의 숫자가 증가할수록 수행시간이 증가하고 출발점과 종착점의 위치에 따라 수행시간이 달라졌다. 그림의 플롯은 시뮬레이션결과를 DXF(AutoCad Drawing Interchange File Format)파일 형식으로 출력하여 문서상으로 바로 읽어 들인 것이다. 이를 위해 선긋기, 원그리기, 다각선 그리기, 점 그리기, 좌표변환, 문자 출력 등의 기본 함수를 포함하는 DXF 라이브러리 함수가 개발 사용 되었다. 시뮬레이션은 각도간격을 3°로 하여 수행하였다.

기하매핑 시간은 작업공간의 복잡도 및 출발점, 종착점의 위치와 관련되고, 이러한 기하매핑 후 후속 처리를 통하여 가장 안전한 궤적, 최단거리궤적등 추가적 요건을 만족시키는 궤적을 연결점공간의 빈 공간으로 부터 검색할 수 있다. 그리고 기하매핑과 검색을 동시에 수행함으로써 충돌회피궤적 검색시간을 단축시킬 수 있다. 또한 각도간격(resolution)을 크게 함으로써, 수행시간을 단축시킬 수 있다. 그림 5 (f)는 연결점공간이 장애물의 이미지로 완전히 투영된 경우를 보여주고, 충돌회피 궤적이 없음을 나타낸다.

6. 고찰 및 결론

주어진 두점사이를 연결하는 장애물과 간섭하지 않는 궤적을 구하는 문제는 상위레벨 로봇제어를 위한 중요한 문제의 하나이

다. 본 논문에서는 이 제적을 공간상의 연결된 두 개의 일차 매개함수로 보는 관점을 도입하였다. 이 두 일차 매개함수를 하나의 연결점에서 연결시키고 나머지 점을 출발점과 종착점에 대응시켜 제적을 형성하였다. 그리고 연결점의 위치를 극좌표로 표현하여 연결점공간을 구성하였다. 이 제적과 각 장애물의 간섭 여부를 연결점 공간상에 투영하여 장애물의 이미지를 구하였고, 이러한 투영작업을 기하매핑이라 정의하였다. 이렇게 하여 유클리드공간 상의 충돌회피 제적계획 문제를 연결점공간의 빈 공간상의 점 선정의 문제로 변환하였다.

기하매핑의 구현을 위해 장애물의 분류작업에 대해 논의하였고, 기하매핑의 성능 등을 검토하기 위해 시뮬레이션을 수행하였다. 시뮬레이션은 PC를 이용하여 수행하였고, 기하매핑 수행시간은 장애물의 숫자와 출발점/종착점의 위치 및 기하매핑에 사용된 변수의(θ) 크기에 따라 변화였다.

7. 참고문헌

[1] Lozano-Pérez, T., "Automatic Planning of Manipulator Transfer Movements," IEEE Trans. Systems, Man, and Cybernetics, Vol. SMC-11, No.10, pp.681-698, 1981
 [2] Kambhampati, S., and Davis, L.S., "Multiresolution Path

Planning for Mobile Robot," IEEE J. of Robotics and Automation, Vol. RA-2, No.3, pp.135-145, 1986
 [3] Khosla, P., and Volpe, R., "Superquadric Artificial Potential for Obstacle Avoidance and Approach," IEEE Proc. Int. Conf. on Robotics and Automation, Philadelphia, pp.1778-1784, 1988
 [4] Lumelsky, V.J., "Effect of Kinematics on Motion Planning for Planar Robot Arms Moving Amidst Unknown Obstacles," IEEE J. of Robotics and Automation, Vol. RA-3, No.3, pp.207-223, 1987
 [5] Bézier, P., Numerical Control Mathematics and Application, Translated by A.R. Forrest and A.F. Pankhurst, John Wiley & Sons, New York, 1972
 [6] Namgung, Ihn, "Planning Collision-Free Paths with Applications to Robot Manipulators," Ph.D. Dissertation, University of Florida, 1989
 [7] Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," Int. J. of Robotic Research, Vol.5, No.1, pp.90-98, 1986
 [8] Shin, D.H., and Ollero, A., "Mobile Robot Path Planning for Fine-Grained and Smooth Path Specifications," Journal of Robotic Systems, Vol. 12 (7), pp 491-503, 1995

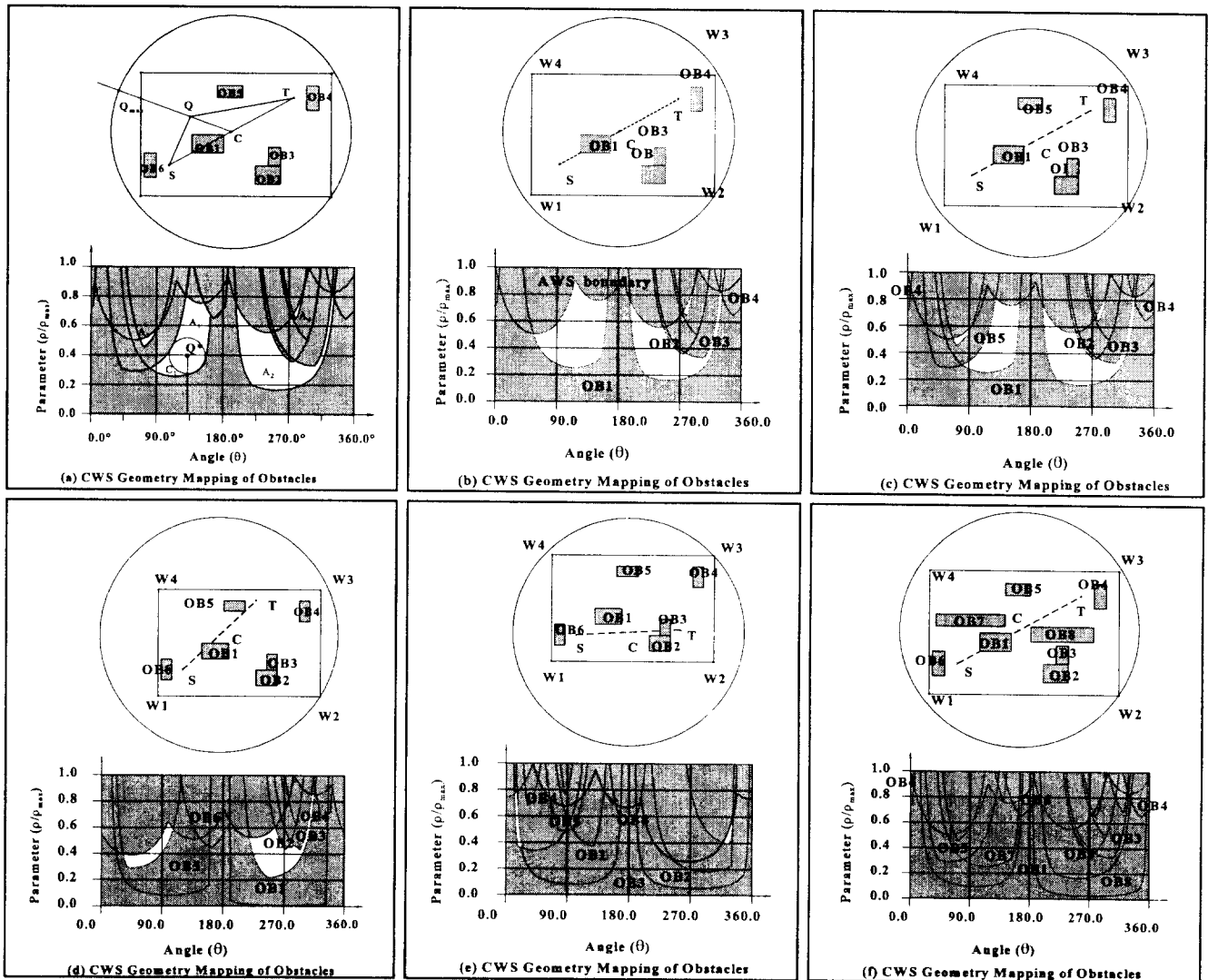


그림 5. 원형 작업공간의 기하매핑 시뮬레이션

Figure 5. Simulation of Circular Workspace Geometry Mapping