

퍼지 집합 이론을 이용한 실린더형 로봇의 3차원 공간상에서의 충돌 회피 운동 계획 시스템
(A Fuzzy Control Application to the collision Avoidance Problem of a Robot)

오 구 관 모
(G.M.Koo)

이 범 희
(B.H.Lee)

고 명 삼
(M.S.Ko)

서울대학교 공과대학 제어계측공학과
로봇릭스 및 지능시스템 연구실

ABSTRACT

This paper investigates the possibility of applying fuzzy algorithms to the collision avoidance problem of a cylindrical robot moving in a 3-dimensional workspace. The displacement information from the end-effector to the points via which the robot can avoid collisions is given by the human operator. Then this uncertain information is adopted for the fuzzy inference system for robot movements. The robot movement is simulated to verify the effectiveness of the proposed fuzzy algorithm.

1. 서 론

환경을 관찰하고 파악하는 인간의 능력은 매우 복잡한 상황을 간단하게 이해한다는 큰 장점을 가지고 있으나, 인간이 감지해 내는 정보에는 필연적으로 불확실성 (uncertainty), 혹은 모호성 (ambiguity)으로 간주될 수 있는 오차가 포함되어 있다.

현재, 이와 같은 인간의 주관적 판단에 의한 정보나 불완전한 계측에 의한 정보에 포함된 불확실성을 처리하기 위해 몇 가지의 이론들이 사용되고 있다.

[15]에서는 이러한 이론들중 확률 이론과 Dempster-Shafer 이론, 그리고 퍼지 셋 (Fuzzy Set) 이론과 전문가 시스템에 사용된 MYCIN/EMYCIN 방법등에 대해 간단히 고찰하고, 그 장단점들을 비교하였다.

본 연구에서는 이들 중 시전 정보를 요구하지 않고 인간의 경험에 의한 직관을 모방하기에 적합하며 빠른 수행속도를 얻을 수 있는 퍼지 집합이론을 이용하여 작업중인 로봇의 충돌 회피 운동을 묘사하고, 그 결과를 분석하였다.

퍼지 집합 이론은 Zadeh [1] 에 의해 처음 소개된 이론으로 초기에는 주로 모델링이 어렵거나 입출력 변수들간의 비선형성이 거저 기존의 제어 방식으로는 만족할 만한 결론을 얻기 어려운 복잡한 공정 제어등에 사용되었다[2]. 이 이론이 로봇릭스 분야에 이용되기 시작한 것은 1980년대 이후로 각 조인트의 서어보 콘트롤과 충돌 회피 문제등의 해결에 주로 응용되었다. Li와 Lau[7]의 논문에서는 DC 서어보 모터 콘트롤에 퍼지 콘트롤을 사용하고 그 결과를 기존의 P, ID 콘트롤 및 MRAC 방법등과 비교하였고, Johnson 과 Feeley[5]는 2축 로봇이 각 조인트 모터의 제어를 독립적으로 수행하면서 수행 평가자의 개념을 도입해 적응성(Adaptiveness)를 갖추었다. Takeuchi 등[6]은 Rotary Crane에 로봇에 의해 부하가 매달려 있는 상황을 묘사하여 모델링이 어려운 복잡한 작업에의 응용 가능성을 제시하였고, Mamdani등은 3축 로봇을 대상으로 물 베이스(Rule Base)의 수정과 삭제, 첨가등이 자동으로 이루어지는 SOC (Self Organizing

System)을 설계, 구현하여 그 실험 결과를 발표하였다[2]. 충돌 회피 문제에서는 인간의 시각 정보를 이용하여 2차원 평면상의 물체를 피해 가는 간단한 실험에 관한 연구가 발표된 바 있다[4].

본 연구에서는 고도의 기능을 가진 센서의 사용이 제한된 상황에서 작업자가 본인의 시각 기능을 이용해 물체에 대한 거리 정보를 주어 작업중인 로봇이 충돌이 없는 경로로 이동하도록 하는 시스템을 구성, 그 동작을 묘사하였다. 대상 로봇은 3축 실린더형 로봇으로 하고 물체는 간단한 다면체로 근사하여 3차원 공간 상에서의 충돌 회피 동작을 구현했다. 이러한 충돌 회피 방식은 종전의 충돌 회피 알고리즘들 [9]-[14]이 대부분 Graph Search 방법을 이용함으로써 상당한 수행시간을 필요로 한다는 단점을 보완하는 효과를 볼 수 있다.

2. 퍼지 추론 시스템의 구성

(1) 퍼지 셋 이론

현실세계에서 어떤 집합이 포함하고 있는 요소들의 성격은 그 포함관계가 명확하지 않은 경우가 많다. 퍼지 집합이란 어떤 원소가 집합에 속하는 정도가 연속함수로 표시되는 집합이다. 퍼지 집합은 각 원소에 0과 1사이의 멤버십 값 (membership grade)을 부여하는 멤버십함수 (membership function) 에 의해 정의되며, 따라서 회원등급이 0과 1의 이진값만을 가지는 일반적인 집합이론에서의 집합에 비해, 보다 광역화된 개념으로 생각될 수 있다.

정의) X를 공간 (space), x를 그 공간상의 원소들이라 가정하면, 공간 X에서의 퍼지 집합 A는 0과 1사이의 값을 갖는 멤버십 함수 f (x)에 의해 다음과 같이 정의된다.

$$A = \{ (x, f_A(x)) \}, \quad x \in X$$

일반적인 집합이론에서 정의되어 있는 각종 연산들은 퍼지집합에서도 모두 정의된다. 특징적이라 할 수 있는 것은 각 퍼지 집합에 정의되어 있는 멤버십 함수들을 이용하여 일반적인 집합이론에서의 집합연산들이 비교연산 또는 산술연산들로 변환된다는 점이다. 예를 들자면 전집합 (universal set) 은 멤버십 함수가 공간 X 상의 모든 원소 x 에 대해 1 의 값을 가지는 집합으로, 공집합 (empty set) 은 멤버십 함수가 X 상의 모든 x 에 대해 0 의 값을 가지는 집합으로 정의된다. 논문에서 사용되는 몇 가지 기본 연산들의 정의는 다음과 같다.

정의)

- 1) 합집합 (union) : 두 퍼지 집합 A 와 B 의 합집합은 $A \cup B$ 로 표기하고, 그 멤버십 함수는

$$f_{A \cup B} = f_A \vee f_B = \text{Max}(f_A, f_B)$$

로 정의된다.

- 2) 교집합 (intesection) : 두 퍼지 집합 A 와 B 의 교집합은 $A \cap B$ 로 표기하고, 그 멤버십 함수는

$$f_{A \cap B} = f_A \wedge f_B = \text{Min}(f_A, f_B)$$

로 정의된다.

- 3) 보집합 (complement) : 퍼지 집합 A 의 보집합은 \bar{A} 으로 표기하고, 그 멤버십 함수는

$$f_{\bar{A}} = 1 - f_A$$

로 정의된다.

이러한 기본연산들외에 본 논문에서는 Mamdani와 Assilian[2] 이 사용한 곱집합(Cartesian Product)과 퍼지 릴레이션 (fuzzy relation) 의 정의를 이용하여 퍼지 추론 (fuzzy inference) 을 수행하였다.

정의) 두 퍼지 집합 A 와 B 의 곱집합은

$$A \times B = \{((x,y), f_{A \times B}(x,y) = \text{Min}(f_A(x), f_B(y)))\} \\ \times C \times X$$

로 정의되며, 퍼지 릴레이션 R 은

$$R = \{((x,y), f_R(x,y)), (x,y) \in A \times B$$

로 정의된다.

(2) 시스템 개요

다음의 그림1은 개발된 시스템의 블록 다이어그램이다. 추론 시스템 (Reasoning System)의 입력은 작업자의 시각으로 관찰한 로봇트 핸드 끝에서 셋 포인트 (Set Point) 까지의 직교좌표계상의 각 축 방향의 상대적 위치이며, 출력은 로봇트가 실제로 이동할 각 축 방향의 변위 및 선속도이다. 그림에서 셋 포인트는 목표 지점 (Goal Position) 까지의 경로에 물체가 없으면 목표 지점으로 지정되고, 물체가 있는 경우는 작업자의 판단에 의해 결정된다. 물 베이스와 함께 추론에 사용되는 세 입력 변수들은 x distance는 x축, y_distance는 y축, height는 z축 방향으로의

로봇트 핸드 끝에서 경유점까지의 상대적 위치이며 물 베이스는 세 변수들의 값에 의해 결정되는 상황에 대한 적절한 로봇트의 운동 계획으로 이루어진다.

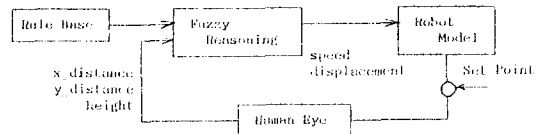


그림 1. Fuzzy Reasoning System

그림 2 는 전체 시스템의 동작을 설명하는 흐름도이다. 엔드 이펙터의 초기 위치와 목표 위치가 입력되면 두 지점에 의해 결정되는 선분의 식과 물체의 각 면의 평면 방정식에 의해 물체와의 충돌 여부가 검사된다. 이 시스템의 목적은 인간이 주는 정보를 처리하는 것이므로 물체의 위치에 대한 정보는 추론에 사용되지 않으나 본 연구에서는 추론 결과의 검증 을 위해 물체의 위치 정보를 사용하였다.

충돌이 감지 되면 작업자는 물체의 모서리 상에서 어느 지점을 경유하여야 충돌이 없이 최단거리로 목표 지점에 도달할 수 있을지를 판단하여 그 점까지의 상대위치를 입력한다. 이 정보는 추론 시스템에 전달되어 물의 입력 변수로 사용된다. 추론에 의해 각 축 방향의 이동거리와 선속도가 결정되면 이 값에 의한 로봇트 엔드이펙터의 운동이 모사되며, 이러한 과정이 목표 위치에 도달될 때까지 계속된다.

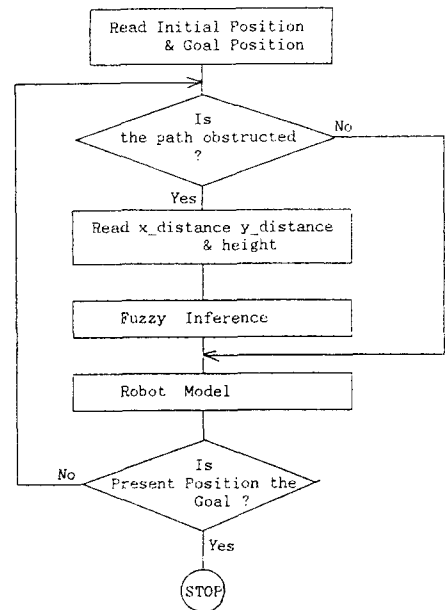


그림 2. 충돌 회피 운동 알고리즘

(3) 퍼지 추론 시스템

추론을 위하여, 로봇트가 한번의 추론으로 도달할 수 있는 각 축 상의 거리를 -5에서 5까지의 11개의 구간으로 나누었다. 즉 작업자는 이 11개의 값 중 실제 거리에 가장 가깝다고 생각되는 값을 입력하는 것이다. 이 11개의 값 위에서 각각 -5, -2, 0, 2, 5를 중심으로 하는 멤버십 함수를 가지는 다섯 개의 계량 변수 NL(Negative Large), NS(Negative Small), ZE(Zero), PS(Positive Small), PL(Positive Large)를 정의하여 추론에 사용하였다. 각 변수에 할당되는 멤버십 함수로는 확률 정규 분포 함수의 모양을 빌어 사용하였다. 이것은 변수에 확률적인 의미를 부여하고자 하는 것이 아니라 확률 정규 분포 함수가 자연계의 많은 현상들의 모델로 적합하다는 점을 감안하여 그 형태만을 사용한 것이다. 실험에서는 기울기(Sharpness)를 달리하는 두 개의 멤버십 함수를 사용하여 그 결과를 비교해 보았다. 표 1과 표 2는 실험에 사용한 두 개의 멤버십 함수 들이다.

표 1. 멤버십 함수 테이블 1

	-5	-4	-3	-2	-1	0	1	2	3	4	5
NL	1	.8	.2	.0	.0	.0	.0	.0	.0	.0	.0
NS	.0	.2	.8	1	.8	.2	.0	.0	.0	.0	.0
ZE	0	.0	.0	.2	.8	1	.8	.2	.0	.0	.0
PS	.0	.0	.0	.0	.0	.2	.8	1	.8	.2	.0
PL	.0	.0	.0	.0	.0	.0	.0	.0	.2	.8	1

표 2. 멤버십 함수 테이블 2

	-5	-4	-3	-2	-1	0	1	2	3	4	5
NL	1	.7	.3	.2	.0	.0	.0	.0	.0	.0	.0
NS	.2	.3	.7	1	.7	.3	.2	.0	.0	.0	.0
ZE	.0	.0	.2	.3	.7	1	.7	.3	.2	.0	.0
PS	.0	.0	.0	.0	.2	.3	.7	1	.7	.3	.2
PL	.0	.0	.0	.0	.0	.0	.0	.2	.3	.7	1

세계의 입력 변수들은 각각 5 개의 계량 변수들로 나누어지므로 가능한 모든 입력 상황의 조합은 하나의 출력 변수에 대해 125 개이다. 실험에 사용한 룰 베이스(Rule Base)는 6개의 출력 변수, 즉 각 축 방향의 이동 거리와 선속도에 대해 23 개씩 총 138 개로 구성되었다. 이것은 수행 속도를 빠르게 하기 위해 실험을 통해 수행에 큰 영향을 주지 않는 물들을 삭제하여 결정된 것이다. 각각의 물은 다음과 같은 형태를 가진다.

" IF (x_distance is A) AND (y_distance is B) AND (height is C) THEN (output is D) "

여기서, A,B,C,D는 NL-PL의 변수들이고, 출력은 각 축 방향의 변위이거나 선속도들이다.

입력이 들어왔을 때 출력 값이 결정되는 과정은 다음과 같다.

"IF (A AND B AND C) THEN D"의 구조는 고 논리적 의미가 " IF (IF (IF A THEN B) THEN C) THEN D "의 구조와 같으므로 $Ax Bx Cx D$ 의 곱집합을 형성한다고 볼 수 있다. 이 때 $A1, B1, C1$ 이라는 퍼지변수가 주어지면 이 때의 출력은 퍼지 릴레이션의 식을 이용하여

$$D1 = \sum_i \text{Max}_{ijk} \{f_A(u_i), f_{A1}(u_i), f_B(v_j), f_{B1}(v_j), f_C(w_k), f_{C1}(w_k), f_D(x_i)\}$$

가 된다.

여기서, 본 논문에서 채택하고자 하는 방법과 같이 세어거의 입력으로 퍼지변수가 아닌 정확한 숫자를 사용하면 f_A, f_B, f_C 은 이 숫자에 대해서만 1 이 되고 나머지 경우에는 모두 0 의 값을 갖게 되어 $D1$ 의 식은 다음과 같이 간단하게 정리된다.

$$D1 = \sum_i \text{Min}\{f_{A1}(u_a), f_{B1}(v_b), f_{C1}(w_c), f_D(x_i)\}$$

(단 u_a, v_b, w_c 는 $A1, B1, C1$ 의 멤버십 함수가 1이 되는 원소)

이렇게 하여 생성되는 퍼지 집합 $D1$ 의 멤버십 함수 값은 물리적으로는 해당 규칙이 모델의 상황에 유효하게 작용하는 정도를 나타낸다고 볼 수 있다.

이러한 추론과정을 통해 결정된 멤버십 함수값과 이 때의 출력 변수값은 디퍼지피케이션을 거쳐 룰로 입력을 생성한다.

디퍼지피케이션(Defuzzification)은 한 개 이상의 물이 적용될 때 실제 적용할 출력값을 계산해 내는 과정으로, 멤버십 값이 최대가 되는 물의 출력을 선택하는 방법과 각 출력 값들을 해당 멤버십 함수 값들에 대해 평균을 내는 방법이 많이 사용되고 있다. 본 연구에서는 추론에 사용되는 입력변수와 출력변수들이 모두 연도 이벡터의 운동거리라는 같은 물리적 의미를 가지므로, 이 두 방법 중 멤버십 값이 최대가 되는 물을 선택하는 방법을 사용하면 입력 축 작업자가 판단한 거리가 그대로 출력되는 경우가 많아 추론의 의미가 없어지게 된다. 따라서, 본 연구에서는 후자의 방법만을 사용하였다. 이를 식으로 나타내면 다음과 같다.

$$I = \sum_{k=1}^n (u_k \times U_k) / \sum_{k=1}^n u_k$$

여기서 u_k 는 각 물의 출력 멤버십 값이고, U_k 는 각 물에서 멤버십 값이 가장 큰 변수값이며, n 은 물의 총 갯수이다.

3. 충돌 회피 알고리즘

이 절에서는 추론을 거친 정보에 의한 로봇의 운동을 모사하는 과정과 주관적 판단에 의한 부정확성을 보상하는 방법에 대해 간단히 기술한다.

(1) 운동의 모사

작업자가 입력한 정보에 의해 추론을 거치면 각 축 방향으로 이동할 거리와 선속도가 출력된다. 이 정보들로부터 직교 좌표계상에서의 경유점들이 계산되고 각 점들에서 역기구학 (Inverse Kinematics) 에 의해 각 조인트 값들이 결정된다. 이를 위해 추론에 의해 결정된 z축 방향의 이동거리와 선속도로부터 '최대 가속-등속-최대 감속'의 속도그래프를 만들어 내며, 이 때 추론의 결과인 z축 방향의 선속도를 속도 그래프에서의 등속부분의 속도로 사용한다. 이동할 로봇의 동력학적 제한 조건은 고려하지 않기로 한다. 선속도를 v_z , 총 이동 거리를 d_z , 상수로 미리 지정되는 최대 가속도를 a_{zm} 이라 하면 속도 그래프는 다음의 과정에 의해 구해진다.

$$\begin{aligned} d_a &= v_z^2 / 2 a_{zm} \\ d_o &= d_z - 2 d_a = d_z - (v_z^2 / a_{zm}) \\ t_a &= v_z / a_{zm} \\ t_o &= d_o / v_z = (d_z / v_z) - (v_z / a_{zm}) \end{aligned}$$

(여기서, d_a 는 가속구간에서의 이동거리
 d_o 는 등속구간에서의 이동거리
 t_a 는 가속하는 데 소요되는 시간
 t_o 는 등속으로 이동하는 시간을 각각 나타낸다.)

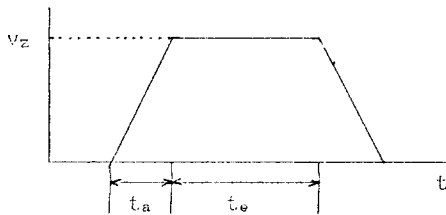


그림 3. z축의 속도 그래프

직교 좌표계에서 결정된 경유점은 로봇의 운동 계획을 위해서 조인트 값 (r, θ, z) 으로 바뀌어야 하는데 이는 다음과 같은 간단한 식으로 결정된다.

$$\begin{aligned} r &= (x^2 + y^2)^{0.5} \\ \theta &= \begin{cases} 0, & x = 0, y = 0 \\ \pi / 2, & x = 0, y > 0 \\ 3\pi / 2, & x = 0, y < 0 \\ \arctan(y/x), & \text{otherwise} \end{cases} \\ z &= z \end{aligned}$$

(2) 충돌 회피

작업자가 입력해 주는 정보에는 축정의 부정확성에 의한 오차 이외에도 지정해준 점 자체가 최소거리 경로가 아닐 가능성도 포함되어 있다. 이러한 종류의 오차를 배제하고 최소거리 경로에 가까운 경로를 선택하기 위하여 로봇의 엔드 이펙터의 시작점에서 본 물체의 모서리상에 7개의 중간 경유점들을 잡아 이 점들을 지나 목표점에 이를 수 있는 거리 정보들을 배열 변수로 입력하여 이들의 코스트 (cost)를 계산, 비교하여 최소 코스트를 가지는 경로를 선택한다. 추론된 경로 중 충돌을 일으키는 경로가 있을 수 있으므로 이를 검사하여 제외한다. 작업자가 판단하는 경유점은 최단 거리 이동을 위한 것이므로 코스트로는 전체이동 거리를 사용했으며, 별도로 성능 향상을 위하여 중간점에서의 물체와의 거리를 코스트로 사용하여 실험해 보았다.

이상의 내용을 단계별로 정리하면 다음과 같다.

- 1) 7개의 경유점 후보지에 대한 거리 정보 입력.
- 2) 위의 점들을 지나는 경로에 대한 코스트 계산.
- 3) 최소 코스트를 가지는 경로에 대해 물체와의 충돌 여부 검사. 충돌이 없으면 이 경로를 선택.
- 4) 충돌이 있으면 제외시키고 남아 있는 경로들에 대해 3)의 과정 반복.

4. 시뮬레이션 결과

시뮬레이션은 IBM/PC-AT 상에서 수행하였으며, 프로그래밍을 위해 사용된 언어는 Turbo Pascal 4.0 이다. 로봇의 엔드 이펙터를 하나의 점으로 모델링하여 그 이동한 경로를 도시하였다.

다음의 그림 4, 그림 5, 그림 6은 각각 이동한 경로를 X-Y, X-Z, Y-Z 평면에서 도시한 것이다.

Initial Position : (0 0 0)
 Goal Position : (180 200 150)

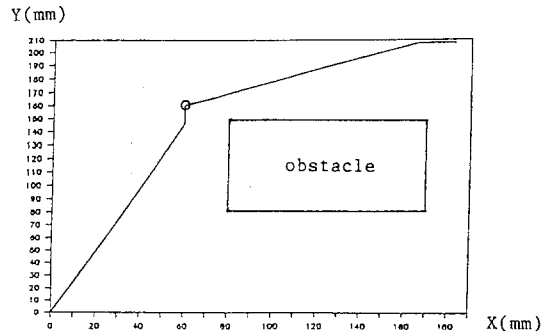


그림 4. X-Y 평면에서 본 충돌 회피 동작

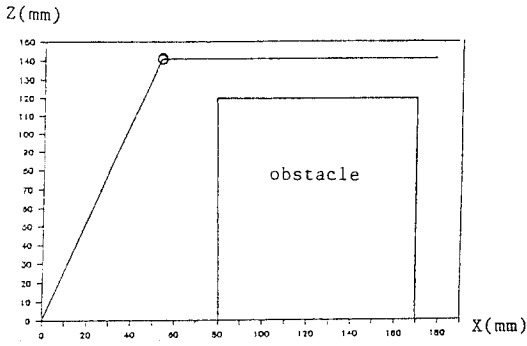


그림 5. X-Z 평면에서 본 충돌 회피 동작

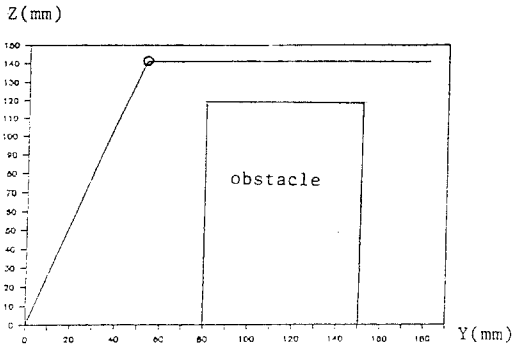


그림 6. Y-Z 평면에서 본 충돌 회피 동작

그림에서 표시된 위치는 엔드 이펙터가 추론의 결과에 따라 이동 후 정지한 곳이다. 이 지점에서 목표 지점까지의 새로운 거리정보를 입력받아 추론을 다시 하여 목표 지점까지 이동한 것이다.

5. 결론

본 연구에서는 퍼지 셋 이론을 도입해 인간의 시각 정보를 이용한 3차원 공간상에서의 로봇의 충돌 회피 운동을 모사하였다. 이는 센서의 사용이 자유롭지 못한 상황에서, Graph Search 방법에 비해 상당히 빠른 속도의 충돌 회피 운동을 구현할 수 있는 방법을 제시하기 위함이었다. 이에 수반되는 오차에 대한 문제는 반복적인 실험에 의한 룰 베이스의 수정이나 경험적 지식의 도입등으로 줄일 수 있을 것으로 예상된다.

참 고 문 헌

[1] L. A. Zadeh, "Fuzzy Sets", Information and control, 8, 1965.
 [2] E. H. Mamdani & S. Assilian, "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller", International Journal of Man-Machine Studies, Vol.7, 1974.
 [3] N. J. Mandic, E. M. Scharf & E. H. Mamdani, "Practical Application of a Heuristic Fuzzy Rule-based Controller to the Dynamic Control of a Robot Arm", IEEE Proceedings, Vol.132, Part D, No.4, July 1985.

[4] K. Nakaishi, T. Matsumoto & Y. Ohta, "An Application of Fuzzy Control to Collision Free Movement Problem of Robot Arm", Japan-U.S.A Symposium of Flexible Automation, pp.103-107, 1986.
 [5] J. C. Johnson & J. J. Feeley, "Adaptive Linguistic Control of a Two Link Robot Arm", Conference on Applied Motion Control, pp.43-46, 1987.
 [6] S. Takeuchi, H. Fujikawa & S. Yamada, "The Application of Fuzzy Set Theory for a Rotary Crane Control", IECON, pp.415-420, 1988.
 [7] Y. F. Li & C. C. Lau, "Application of Fuzzy Control for Servo Systems", IEEE Conference on Decision & Control, 1988.
 [8] G. Vachtsevanos & H. Hexmoor, "A Fuzzy Logic Approach to Robotic Path Planning with Obstacle Avoidance", Proc. of 25th Conference on Decision and control, December 1986.
 [9] T. Lozano-Perez & M. A. Wesley, "An Algorithm for Planning Collision Free Paths among Polyhedral Obstacles", Communications of the ACM, Vol.22, No.10, Oct. 1979.
 [10] N. J. Nilsson, "A Mobile Robot Automation, An Application of AI Technique", Proc. Int. Joint Conf. on AI, 1969, pp.509-520.
 [11] J. W. Boyse, "Interference Detection Among Solids and Surfaces", Communications of the ACM, Vol.22, No.1, pp.3-9, Jan. 1979.
 [12] J. L. Bentley & T. A. Ottmann, "Algorithms for Reporting and Counting Geometric Intersections", IEEE Trans. on Computer, Vol.C-28, No.9, pp.643-647, Sept. 1979.
 [13] S.M.Udupa, "Collision Detection and Avoidance in Computer Controlled Manipulators", pp.737-748.
 [14] J. W. Roach & H. N. Boaz, "Coordinating the Motions of Arms in a Common Workspace", IEEE J. Robotics and Automation, Vol. RA-3, No.5, pp.437-444, Oct. 1987.
 [15] S. J. Henkind & M. C. Harrison, "An Analysis of Four Uncertainly Calculi", IEEE Trans. on SMC, Vol.13, No.5, 1988.