

# 지연시간 퍼지제어기를 이용한 이동로봇의 장애물 회피

유영순\* · 가춘식\*\*

## Use of the Delayed Time Fuzzy Controller for Obstacle Avoidance of Mobile Robot

Yeong Soon Ryu, Chun Sik Ga

**Key Words:** Obstacle Avoidance(장애물 회피), Fuzzy Controller(퍼지 제어기), Delayed Time(지연 시간).

### Abstract

This paper presents a delayed time path planning method of the Autonomous Mobile Robot using fuzzy logic controller for avoidance of obstacles in unknown environment. It is the objective of this paper to develop fuzzy control algorithms using delayed time techniques to deal with moving obstacles randomly. This control method gives the benefit of the collision free movement in real time and optimal path to the pre-settled goal. The computer simulations are demonstrated the effective of the suggested control method in obstacle avoidance.

### 1. 서론

장애물 회피에 대한 연구는 최근 산업자동화에 로봇의 사용이 증가함으로 인하여 사람의 역할을 대신할 이동 로봇의 등장과 함께 더욱더 활발해지고 있다. 이의 기본 개념은 장착된 센서로부터 진행 방향에 놓여진 정보를 입력받아 이를 근거로 고정 또는 이동 장애물을 피하면서 도달하고자 하는 목표점까지 최적 경로로 진행하는 과정이다.

고정 장애물에 대한 연구로는 최단 경로 기본 주행에서 시작되어, 다양한 형상에 대한 회피에

서 최종적으로 국부점 문제에까지 거의 가능한 모든 경우에 대하여 완전한 해법이 개발되고 있는 실정이다. 그러나 이동 장애물의 경우는 장애물의 위치가 실시간으로 변함으로 인하여 이전 상황의 자료를 분석하여 현 상황의 판단 자료로 사용하는 제어 기법으로는 능동적으로 대처하는데 많은 문제가 따르고 있다. 이러한 문제의 해결을 위하여 Kant, Brooks, Fiorini, Haral 및 Maeda<sup>(1-5)</sup> 등의 많은 연구자들에 의해서 다양한 주제들을 가지고 연구가 진행되어 왔다. 하지만 거의 대부분의 논문에서 이동 장애물을 회피하기 위한 완전한 기법은 제시되지 않았고 대다수의 연구는 이러한 문제를 보상하기 위한 방법으로 이동 장애물에 대한 한정적인 가정과 함께 ad hoc 적 기법이 적용되고 있다.

본 논문에서는 랜덤하게 움직이는 장애물에 대

\* 안동대학교 기계공학부

\*\* 안동대학교 기계공학과 대학원

해서 실시간으로 유연하게 회피 동작을 할 수 있도록 지연 시간 기법을 제안하였으며 이를 모의 실험을 통하여 고정 과 이동 장애물에 적용하여 성능을 증명하였다.

## 2. 시간 지연 기법

이동 장애물에서는 장애물이 랜덤하게 움직이므로 그 방향과 속도를 예측하여 실시간으로 제어하기가 쉽지 않다. 이전 단계의 동작에서 획득된 정보로 추후 경향을 판단하는 방법은 실시간 제어에서는 적합하지 않다. 이러한 문제점을 보완하기 위해서 본 논문에서는 시간 지연 기법을 적용한 퍼지제어기를 제안하게 되었다. 이 기법은 제어 동작을 수행하는 전체 시간을 분할해서 수행하는 시간의 처음 일정 부분의 시간이 지난 후에 경향을 확인하여 그 수행시간 내에 판단 알고리즘을 세워서 실시간으로 로봇이 움직이게 된다. Fig. 1에서 장애물은 이전 1초간 진행한 후 다음 1초 동안 Fig. 1의 상단과 같이 방향을 수정 진행하였다. Fig. 1에서 로봇이 이전 1초간의 데이터만을 사용하여 판단하였을 경우에는 충돌이 발생하게 되지만 시간 지연 기법을 이용하여 처음 0.1초가 지난 후 판단을 한 경우에는 장애물의 최신 진행 방향 정보를 받을 수 있어 충돌을 피할 수 있게 됨을 알 수 있다. 물론 0.1초라는 초기값 손실은 있으나 이것은 분할 정도를 줄임으로써 보상될 수 있다.

Fig. 2는 시간 지연이 없는 기존의 방법과 수행 기간(1초)을 시분할 하여 처음 0.1초 구간에서는 제어 동작 없이 장애물의 이동 경향을 판단하고 나머지 0.9초 동안에서는 제어 동작이 이루어지는 시간 지연 기법을 비교한 시간 진행도이다. Fig. 3은 퍼지제어기에서 시간 지연 관계를 나타낸 것이고 이는 간단히 식 (1)과 같이 나타낼 수 있다.

$$u(t+\tau) = f(d(t), \theta(t), l(t)) \quad (1)$$

$u$  : 퍼지 제어기 출력

$d$  : 이동 로봇의 이동 거리

$\theta$  : 로봇 진행방향과 장애물이 이루는 각

$l$  : 장애물과 이동 로봇과의 거리

$f$  :  $d, \theta, l$  에 의한 함수

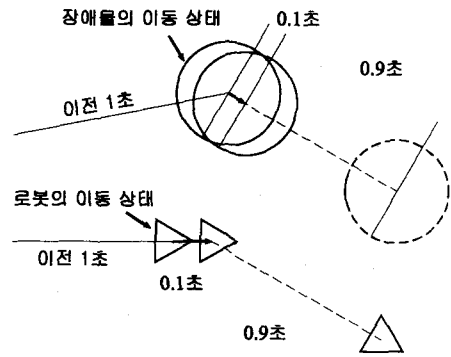
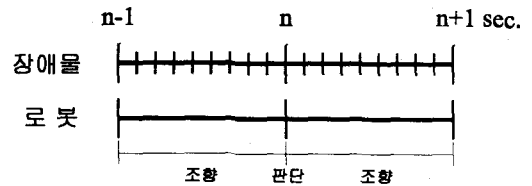
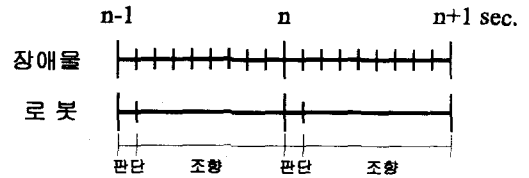


Fig. 1 Geometric diagram of delayed time technique between obstacle and robot



(a) Relationship between obstacle and robot in the step time domain



(b) Suggested delayed time technique in the step time domain

Fig. 2 Robot motion in the step process



Fig. 3 Delayed time fuzzy controller

## 3. 상대속도 계산

Fig. 4는 상대 속도 계산을 위한 도해이다. 그림에서  $l_1$ 과  $l_2$ 는 임의의 시간 스텝  $n$ 과  $n+1$ 에서 센서를 통해 얻어진 장애물과 로봇과의 최소 거리이고,  $\theta_1$ 과  $\theta_2$ 는 로봇과 장애물이 이루는

벡터와 로봇 진행방향과 이루는 각도이다.  $V_x$  는 장애물의  $x$  방향 진행 벡터이고  $V_y$  는

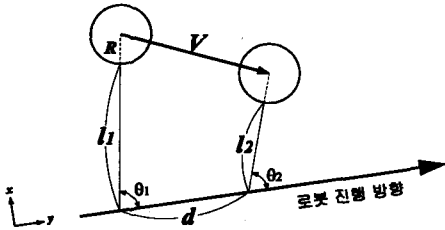


Fig. 4 Diagram for relative speed calculation

장애물의  $y$  방향 진행 벡터를 나타내며,  $d$ 는 한 스텝에서 로봇이 진행한 거리를 나타낸다. 여기에서 각각의 상대속도는 아래와 같이 계산될 수 있다.

$$V_x = (l_2 + R) \sin \theta_2 - (l_1 + R) \sin \theta_1$$

$$V_y = (l_2 + R) \cos \theta_2 - (l_1 + R) \cos \theta_1$$

#### 4. 센서 정보 추출

Fig. 5는 로봇에 장착된 센서의 배치도를 보이고 있다. 센서는 일반적으로 거리 측정에 사용되는 초음파 센서로 가정하였고 센서의 개수는 계산의 편리성과 유효성을 감안하여 총 7개이고 Fig. 5와 같이  $30^\circ$  도의 간격으로 배치했다. 각각의 센서는 실제와 유사하게 측정각을  $10^\circ$  로 지정하였고 유효 센싱 범위는 20cm로 지정했다. 로봇의 진행 방향을 기준으로 좌·우  $90^\circ$  씩  $180^\circ$  를 센싱하도록 하였다. 센서 개수의 제한으로 센싱 범위 안으로 들어와도 센싱이 되지 않는 경우가 발생할 수 있으므로 장애물의 크기를 직경 20cm 로 가정하였다. 이는 센서의 개수와 오차 범위에 대한 문제로 상황에 따라 최적값을 지정할 수 있다.

#### 5. 회피 주행 알고리즘

회피 주행을 위한 알고리즘은 Fig. 6의 순서도와 같이 나타낼 수 있다.

Step.1에서는 로봇이 센서를 통해 주위 장애

물을 감지하며 로봇과 장애물과의 상대 속도는 Step.2에서 계산되어진다. Step.3에서는 퍼지

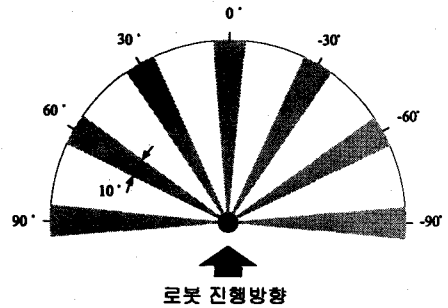


Fig. 5 Sensor array

주행 회피를 위해 획득된 데이터를 이용하여 퍼지 입력을 생성한다. Step.3에서는 장애물이 위치한 영역에 따라 회피하는 방법이 다르고, 이동 장애물인 경우에는 진행 방향의 후방을 목표로 잡았다.

회피 동작은 1초 간격으로 이루어지되 처음 0.1초 동안은 이전의 회피 동작이 연속적으로 이루어지면서 동시에 상대속도를 검출하여 나머지 0.9초 동안 회피 동작이 이루어지도록 하였다.

Fig. 7은 퍼지입력 생성을 위한 설정도를 보이고 있다. 그림에서 영역을 3개의 경우로 분할하였으며 각 영역에서 회피하는 방법은 아래의 제어 규칙을 따르도록 하였다.

- R1 : 영역 1에서  $V_x=0, V_y<0$  인 경우,  
도착지와 가장 가까운 방향으로 회피.
- R2 : 영역 2에서  $V_x<0, V_y<0 (L)$   
 $V_x>0, V_y<0 (R)$  인 경우,  
장애물이 접근하는 쪽으로 회피.
- R3 : 영역 3에서  $V_x<0, V_y>0 (L)$   
 $V_x>0, V_y>0 (R)$  인 경우,  
속도를 줄여서 회피.
- R4 : R1, R2, R3에 해당되지 않는 경우,  
회피동작을 하지 않는다.

#### 6. 퍼지 제어기 모델

장애물 회피와 목표점 접근을 위한 퍼지제어기에 Mamdani 추론법을 사용하였고 무게 중심법

을 이용하여 비퍼지화 과정을 수행하였다. 퍼지 시스템의 구성은 Fig. 8과 같다.

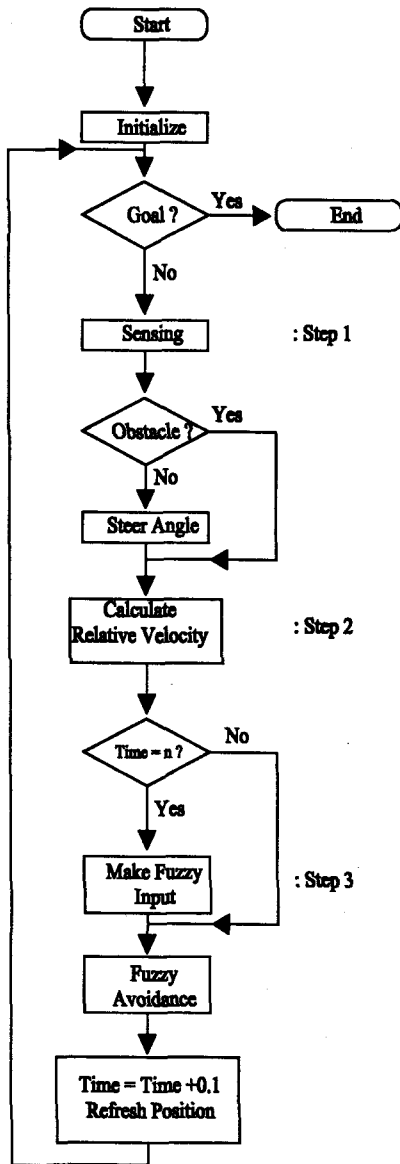


Fig. 6 Flow chart for the fuzzy control avoidance algorithm

이동 로봇의 주행 시 환경인식을 위해 측정된 입력변수의 값을 먼저 언어적인 변수로 나타내기 위해 소속함수를 이용한 퍼지화 과정이 필요한

데, 여기에서는 삼각형 형태의 함수를 이용하여 퍼지화를 수행하였다. 퍼지 제어기에 사용되는 언어변수는 다음과 같다.

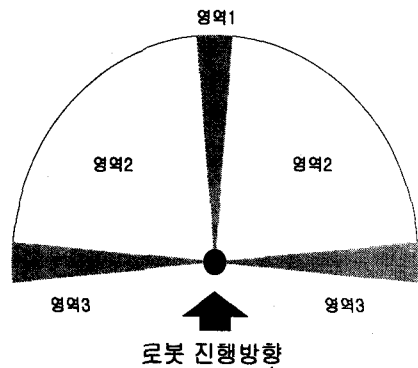


Fig. 7 Diagram for the selected fuzzy input



Fig. 8 I/O relationship of fuzzy controller

- RB(Right Big), RS(Right Small)
- LS(Left Small), LB(Left Big)
- ZZ(Zero), ME(Medium)
- SL(Slow), FA(Fast)
- NR(Near), EM(Emergency)

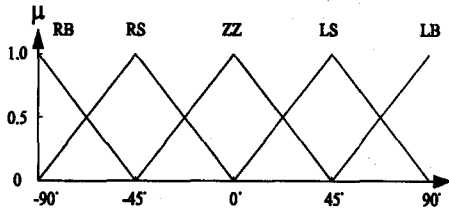
위에 설정된 언어변수로 만들어진 입출력 변수에 대한 소속함수는 Fig. 9에 나타내었다.

주행 과정에서 속도와 조향각을 조절하기 위한 퍼지 제어기의 규칙은 Table 1 과 같다.

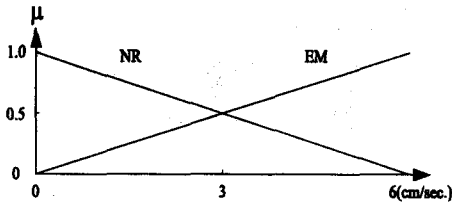
## 8. 모의실험 조건 및 결과

고정 및 이동 장애물에 대해 고려해 볼 수 여러 조건 중 대표적인 3가지 경우에 대한 모의실험 조건 과 주행 결과는 Table 2와 같다. 사용한 환경은 100cm × 100cm의 직사각형 실내, 최대 속도는 3cm/sec, 장애물 반경은 10cm, 로봇의 출발점은 (0,0) 그리고 로봇의 도착점은 (100,100)으로 하여 세 가지 경우에 대해 모의 실험을 수행하였다. 첫 번째 경우에는 고정 복수 장애물을 회피하는 경우로써 총 경로 거리는 약

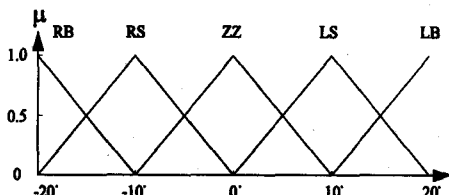
160.06cm 이고 소요 시간은 50.0sec이다. 모의실험 결과는 Fig. 10에서 보여진다. 두 번째 경우는



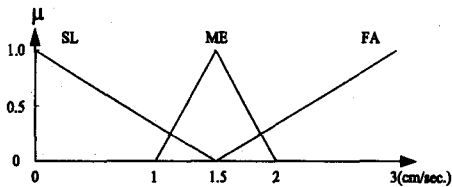
(a) Avoidance angle



(b) Relative speed between robot and obstacle



(c) Steering angle



(d) Steering speed

Fig. 9 Membership function

고정 이동 복합 장애물의 경우로써 고정 장애물은 (40, 30)에 위치하고 이동 장애물은 (0, 80)에서 속도 2cm/sec, 방향 0°로 진행하다가 15초 후 -45° 방향으로 이동을 한다. 총 경로거리는 약 165.33 이고 이때 걸린 시간은 50.7sec 이다. 모의실험 결과는 Fig. 11에서 보여진다. 세 번째 경우는 두 개의 이동 장애물을 회피하는 경우로써 첫 번째 장애물은 (0, 40)에서 속도 3cm/sec, 방향 0°로 진행하다가 5초 후 -45° 방향으로 이

Table 1 Fuzzy control rule base

Fuzzy Input		Fuzzy Output	
Angle	Relative Velocity	Steering Angle	Velocity
RB	NR	RB	ME
RB	EM	RB	SL
RS	NR	RS	FA
RS	EM	RS	ME
ZZ	NR	ZZ	FA
ZZ	EM	ZZ	SL
LS	NR	LS	FA
LS	EM	LS	ME
LB	NR	LB	ME
LB	EM	LB	SL

동을 하고 두 번째 장애물은 (80, 100)에서 속도 2cm/sec, 방향 -130°로 진행하다가 20초 후 -90° 방향으로 이동을 한다. 모의실험 결과는 Fig. 12에서 보여진다.

고정 장애물의 경우에서 회피동작이 큰 이유는 회피 후 목표점 진행 시에도 퍼지 제어에 의해 회피가 가능하도록 되어있는 구조로 인한 것이다. 이는 센서 범위 내에 장애물이 인지되지 않아야만 목표점을 향해 가는 방식이다. 고정 장애물의 경우는 불필요하나 상시 계측으로 인한 이동 장애물의 경로 변경을 인지할 수 있는 장점이 있다고 하겠다. 위 결과에서 보여주듯이 시간 지연 기법을 적용한 퍼지제어기의 경우 고정 및 이동장애물의 경우에서 장애물의 이동 방향의 변화에 대해 원만한 주행이 이루어짐을 확인하였다.

Table 2 Parameters for the simulation and results

	case 1	case 2	case 3
	고정 복수 장애물	고정 이동 복합 장애물	이동 복수 장애물
장애물 특성	각 장애물의 중심점 O1 : (40,30) O2 : (20,70) O3 : (80,20) O4 : (60,80)	고정 장애물의 중심점 (40,30)	첫 번째 장애물 시작점:(0,40) 속도: 3 cm/sec 방향: 0° → -30°
		이동 장애물 시작점:(0,80) 속도: 2 cm/sec 방향: 0° → -45°	두 번째 장애물 시작점:(80,100) 속도 2 방향:-130° → -90°
총 경로 거리	160.06 cm	165.33 cm	157.05 cm
총 경로 시간	50.0 sec.	50.7 sec.	46.9 sec.

## 9. 결론

본 논문에서는 실시간으로 변하는 이동 장애물을 회피하기 위하여 새로운 시간지연 기법을 적용한 퍼지 제어기를 개발하였으며 이의 성능을 증명하기 위하여 고정과 이동 장애물에 대하여 적용하여 다음과 같은 결론을 얻었다.

1) 고정 장애물의 경우, 고정 장애물은 이동 장애물의 특별한 경우로 이해할 수 있으며 이 경우 제안한 기법을 적용하여 안전한 회피를 할 수 있었다

2) 이동 장애물의 경우, 고정과 이동이 동시에 주어졌을 경우와 2개 이상의 이동 장애물의 경우로 나누어서 이 기법을 적용하였으며 모두 실시간으로 장애물의 예측치 못한 변화에 충돌 없이 유연하게 반응함을 알 수 있었다.

## 참고문헌

- (1) K.Kant and S.W. Zucker, 1986, "Toward efficient trajectory planning : The path-velocity decomposition.", The International Journal of Robotic Research, vol. 5, pp.72~89.
- (2) R.A.Brooks, 1986, "A robust layered control system for a mobile robot", IEEE J. Robotics and Automation, Vol. 2, No. 1, pp14~23.
- (3) P. Fiorini and Z. Shiller. 1993, "Motion planning in dynamic environments using the relative velocity paradigm.", Proc. IEEE int. conference on Robotics and Automation, vol.1 pp.556~560.
- (4) K. Shimada, S. Mabuchi and K. Hara, 1991, "Identification of operators judgement rules of danger in collision avoidance maneuvering of ships," Proc. 7th Fuzzy Syst. Symp., Japan, pp.509~512.
- (5) Y. Maeda, and T. Yamanaka, 1990, "Fuzzy obstacle avoidance control with the strategy of operator obtained by the CMAC learning." Proc. 6th Fuzzy Syst. Symp., Japan, pp.531~534.

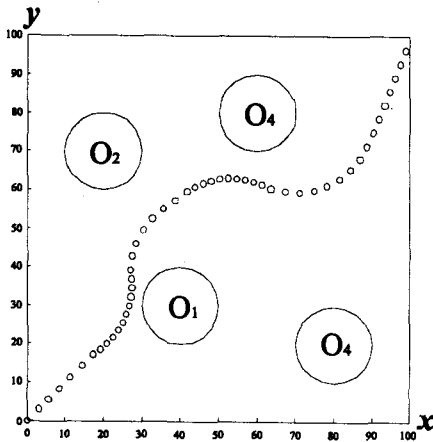


Fig. 10 Simulation result for the case 1

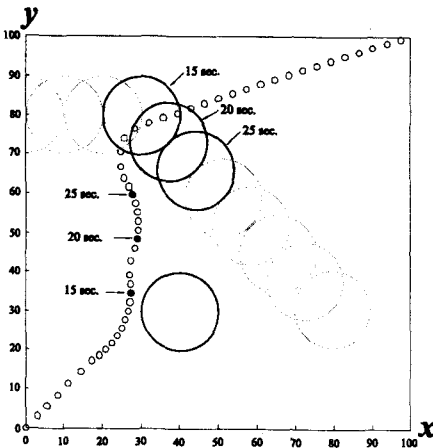


Fig. 11 Simulation result for the case 2

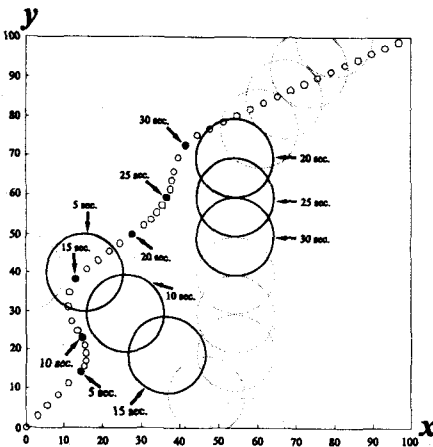


Fig. 12 Simulation result for the case 3