

원격제어 이동로봇의 효율적 주행제어

Efficient navigation control of a Remote Controllable Mobile Robot

정지봉, 이상식, 신위재

Ji-bong Jung, Sang-sik Lee, Wee-jae Shin

요 약

본 논문은 FLC(Fuzzy Logic Control)를 이용한 원격제어 이동로봇이 다수 경유점을 보다 효율적으로 주행하는 방법을 연구하였다. 제작된 원격 제어 이동로봇은 전진, 후진, 좌회전, 우회전의 이동 명령을 한 가지씩 수행한 후 정지한다. 이러한 주행형태의 단점을 보완하기 위해 본 논문은 센서를 이용하여 얻어진 지형정보를 바탕으로 목적지까지 경유점 수를 줄이고, 경유점 간의 이동을 최적한 경로로 주행하도록 하였다. 그리고, 장애물 출현 시 회피할 수 있는 FLC 알고리즘을 개발하였다. 그리고, 이러한 주행 알고리즘을 시뮬레이션을 통해 확인하였다.

ABSTRACT

In this paper, we study how the remote controllable mobile robot which could come to many via points with FLC(Fuzzy Logic Control) efficiently. The fabricated robot stop after the movement of single path method by four kinds of commands (forward, backward, turn left, turn right). To reduce disadvantages of this driving type, this paper reduce via points to goal position base on map which get from sensor, let robot drive via point to via point on optimized path. An algorithm for the avoidance of unexpected obstacles by FLC is developed. And these algorithms are confirmed by computer simulations

I. 서 론

이동로봇은 로봇과 분리되어 있는 중앙 통제로부터의 명령에 의해 움직이는 원격제어용 이동로봇과 자율적인 판단으로 운행하는 자율이동로봇으로 구분할 수 있다. 이 두 가지 형태의 로봇 모두 외부환경을 인식하기 위해 주행구역에 대한 지도를 바탕으로 주행을 한다.

이동로봇의 경로계획은 정보에 따라 전역경로계획(Global path planning)과 지역 경로계획(Local path planning)이 있다.

전역경로계획은 지식기반 경로계획(knowledge based path planning)이라고도 하며 이동하기 전에 고정된 장애물의 정보를 가지고 최적의 경로를

를 찾을 수 있지만 돌발적인 환경변화에 대해서는 개루프 특성 때문에 적용이 어렵다

지역 경로계획은 장애물에 대한 사전지식 없이 센서의 데이터를 기반으로 스스로 판단하여 주행하는 것으로 변화하는 환경에 대해 적응성은 있으나 최적의 경로를 찾기는 어렵다.

따라서, 이동로봇이 주행시 돌발적인 장애물에 대해서는 전역경로 계획으로 대처하기 어렵기 때문에 지역 경로계획이 필요하다.

일반적으로 자율이동로봇은 전역 경로계획을 주어진 알고리즘대로 최적의 경로를 계산하여 주행하게 된다

그러나, 제작된 원격제어 이동로봇은 초음파 센서를 이용해서 얻은 지도를 바탕으로 전역경로계획이 세워지며 로봇 조종자에 의해 지역경로계획

은 전진, 후진, 좌회전, 우회전의 이동명령에 의해 이루어지므로, 목적지까지의 경유점 수를 줄일 수 없고 따라서, 목적지까지 최단경로로 주행한다고 볼 수 없다.[1]

따라서, 본 논문에서는 경유점까지의 최단거리를 이동로봇과 경유점을 일직선으로 주행하는 것으로 정의하고, 경유점까지 최단거리로 자율주행 중 돌발적인 장애물 출현시 장애물을 회피한 뒤 경유점까지 주행하는 형태로 주행 알고리즘을 연구 하였고 주행형태에 대한 입출력 데이터를 측정 수집하여 이동로봇의 주행 형태를 모델링 한 후 이 모델을 퍼지 제어 규칙으로 사용하였다.

II. 지역경로계획

제작된 원격제어 이동로봇은 한 개의 초음파센서가 360° 회전하면서 측정하기 때문에 초음파센서의 난반사와 같은 몇가지 단점이 있어도 정확도가 비교적 우수하고 가격이 저렴하고, 하드웨어 구성이 간단해서 초음파센서를 이용해서 환경을 인식하고 환경지도를 만든다.[2] 그리고, 본 논문에서는 이 지도를 바탕으로 출발점에서 원하는 목적지까지 최적의 경로를 조정자가 생성하고 경유점을 xy 좌표값으로 입력하였다.

지역경로계획은 작성된 경로를 이용, 이동로봇이 주행시 환경인식 센서에서 얻어지는 정보를 FLC(Fuzzy Logic Control)를 이용해서 이동로봇의 속도와 방향을 제어하며, 경유점까지 부드럽게 주행하면서 장애물 출현시 회피하도록 하는 지역 경로계획의 주행 형태를 <그림1>에 나타내었다.

<그림1>에서 주행중 벽면과 가까워졌을 때 벽과의 거리정보를 바탕으로 FLC를 이용해서 θ 만큼 이동 방향각을 조정하고 감속한다.

이동로봇이 <그림2>에서처럼 주행방향으로 좌우45도 범위 내에 있는 물체만 장애물로 인식하고 좌·우45도 밖의 물체는 주행방향에 영향을 주지 않으므로 벽으로 인식하도록 하였다. 그리고 장애물 인식은 전방에 부착된 레이저 센서 F1과 F2에 의해 좌·우 45도 사이에 100cm거리로 하였다.

<그림3>은 이동로봇이 장애물이 있을 때 회피하는 것과 이동로봇의 입출력 변수를 나타내었다

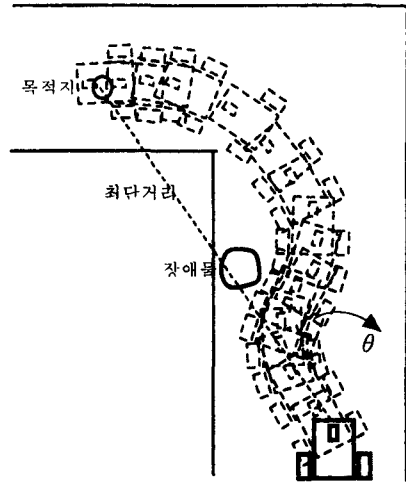


그림 1 . 이동로봇의 지역경로계획
fig 1. A local path planning of the mobile robot

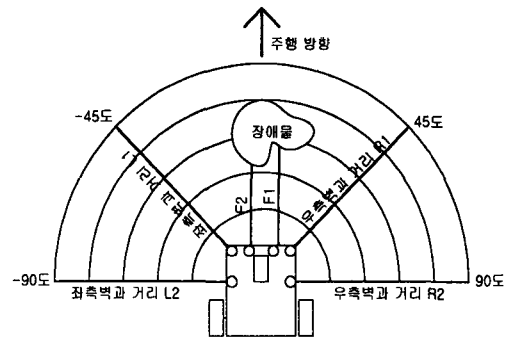


그림 2. 이동로봇의 장애물 인식
fig 2. An obstacle recognition of the mobile robot

.전방센서 F1,F2에 의해 장애물로 인식이 되면 회피 방향은 좌·우측벽 거리정보 L1과R1을 비교하여 긴 거리 쪽을 회피방향으로 정하고 같은 경우는 L2와 R2를 비교하고, 또 같은 경우 목적지 쪽으로 방향각을 계산하도록 하였다.

<그림4>는 앞에서 언급한 지역경로계획에 대한 순서도를 나타내었다. 주행중 전방센서에 의해 장애물을 판별해서 장애물이 아니면 FLC1과 FLC2에 의해 목적지까지의 방향과 속도를 계산하여 주행하고, 장애물로 판단하면 FLC3에 의해 장애물 회피각을 계산하고 20cm/s로 감속 주행을 하도록 하였다

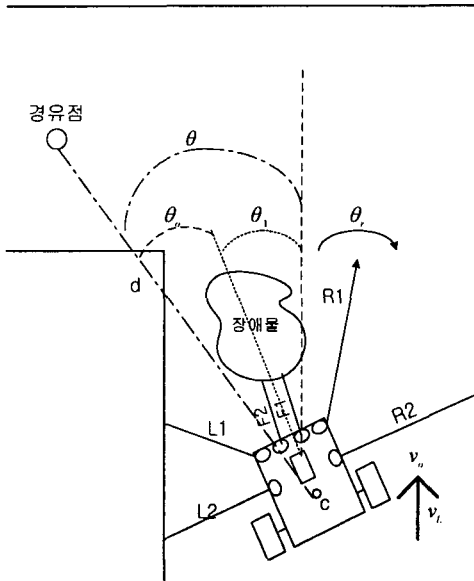


그림 3. 이동로봇의 장애물 회피 및 입출력 변수
fig 3. Avoidance of an obstacle and variables of input and output

c : 이동로봇의 중심

입력변수

- θ_1 : 이동로봇의 지향각
- θ : 로봇지향각과 경유점과의 각도
- d : 이동로봇과 경유점과의 거리
- R_1, R_2 : 이동로봇과 우측벽과의 거리
- L_1, L_2 : 이동로봇과 좌측벽과의 거리
- F_1, F_2 : 이동로봇과 장애물과의 거리
- v_L : 이동로봇의 현재속도

출력변수

- θ_r : 장애물 회피각도
- θ_o : 경유점까지 최단거리 주행 각도
- v_o : 이동로봇의 속도

III. 원격제어 이동로봇의 퍼지 모델링

퍼지 제어규칙 작성은[3] 일반적으로

- i) 제어전문가의 지식을 활용하는 방법
- ii) 숙련자의 조작행위를 모델링하는 방법

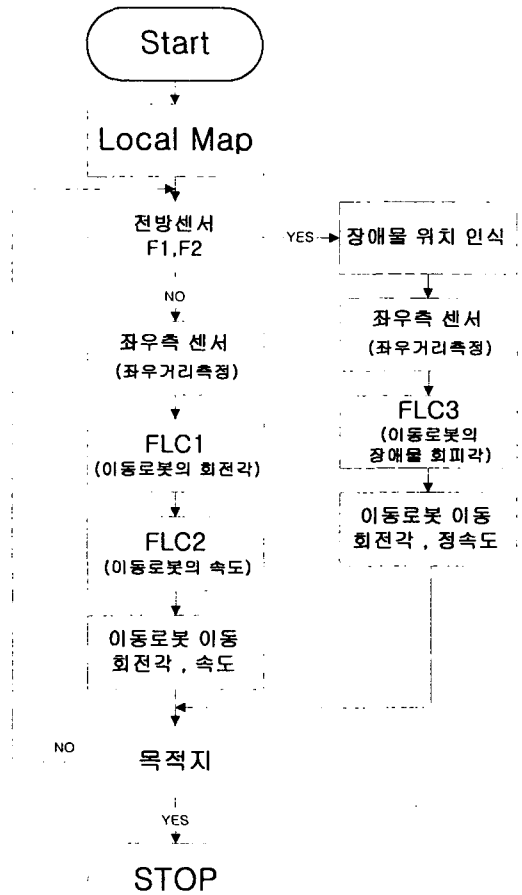


그림 4. 지역경로계획의 순서도

fig 4. Flow chart of local path planning

iii) 제어대상의 퍼지모델을 기반으로 하는 방법

iv) 학습에 의해 제어규칙을 작성하는 방법 등을 복합적으로 사용하고 있다.

본 논문의 이동로봇은 제어전문가의 지식을 이용하여 퍼지 제어규칙을 작성하였다.

앞에 언급한 지역경로계획의 이동로봇 주행형태를 다시 정리하면,

- i) 경유점과 로봇의 지향각이 직선이 되도록 로봇 지향각을 조정한다.
- ii) 경유점과 거리가 멀 경우 현재속도보다 높이고 가까울 경우 속도를 낮춘다.
- iii) 주행시 좌·우측벽면과 가까울 경우 속도를 줄이고 부딪히지 않게 이동로봇의 지향각을 조정한다.
- iv) 장애물이 있으면 회피하고 감속 주행한다.

iv) 이동로봇의 속도는 안전한 주행을 위해 급속한 가속을 하지 않는다.

위의 주행 형태를 모델링 한 후 이 모델을 퍼지 제어규칙으로 사용하였고 이동로봇의 입출력 변수는 <그림3>에 나타내었다.

<그림5>는 이동로봇의 주행 알고리즘에 대한 FLC의 입출력변수에 대해 나타내었다.

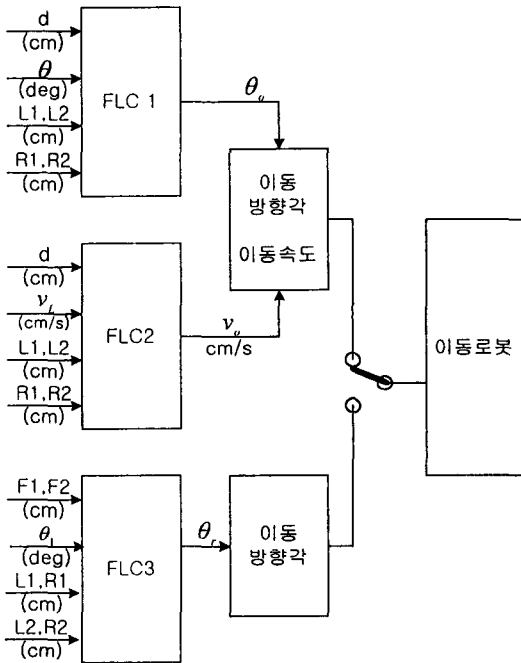


그림 5. FLC의 입출력 변수
fig 5. Input-output variables of FLC

FLC1은 L_1, L_2 와 R_1, R_2 에서 가장 짧은 거리와 경유점과의 거리 d , 경유점과 방향각 θ 를 입력 데이터로 이동로봇의 이동 방향각 θ_0 를 계산한다.

FLC2는 L_1, L_2 와 R_1, R_2 에서 가장 짧은 거리와 경유점과의 거리 d , 이동로봇의 현재속도 v_L 을 입력 데이터로 이동 로봇의 이동 속도 v_0 를 계산한다.

FLC3은 장애물과 이동로봇간의 거리 F_1, F_2 와 이동로봇의 현재 지향각 θ_1 그리고 좌·우측 벽과의 거리를 입력받아 장애물 회피 방향각 θ_2 을 계산한다.

<그림6>은 경유점과 이동로봇과의 거리 (d)에 대한 소속함수를 나타내었고 다음과 같이 정의하였다.

NE : near, MN : medium near, ME : medium MF : medium far, FA : far

<그림7>은 경유점과 이동로봇의 지향각과의 각도 (θ)에 대한 소속함수와 이동로봇의 출력 회전각 (θ_0)에 대한 소속함수로 동일하게 나타내었다.

NB, NM, NS, ZE, PS, PM, PB

N : negative, P : positive, ZE : zero

B : big, M : medium, S : small

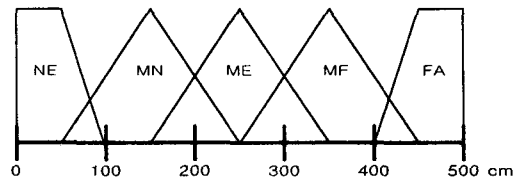


그림 6. 경유점과 이동 로봇과의 거리관계에 대한 소속함수

fig 6. Membership function of the distance relations between mobile robot and via points

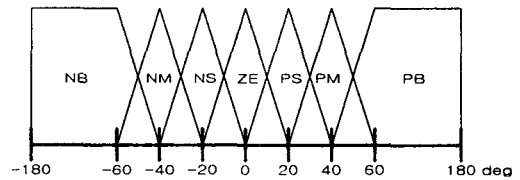


그림 7. 경유점과 이동 로봇의 지향각과의 각도 및 이동로봇의 회전각 출력에 대한 소속함수
fig 7. Membership function of angle from the heading of robot to the target point, and the output rotation angle of the robot.

<그림8>은 이동로봇과 좌·우측 벽과의 거리에 대한 소속함수로 좌·우 동일하게 정의하였고, 주행 중 좌·우측 벽에 접근하면 속도를 줄이고 일정거리를 유지하도록 한다.

<그림9>는 이동로봇의 현재속도(v_L)에 대한 소속함수와 FLC2의 출력속도(v_0)에 대한 소속함수로 동일하게 나타내었다. 최대속도를 60cm/s로 정의하였고 중간속도구간을 넓게 해서 최대속도

로 주행하는 경우를 줄이는 형태로 구성했다.

SL : slow , MS : medium slow

MFS : medium fast , VFS : very fast

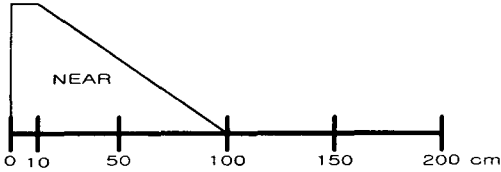


그림 8. 좌·우측벽과 이동로봇과의 거리에 대한 소속함수

fig 8. Membership function of distance relation between robot and left·right side wall

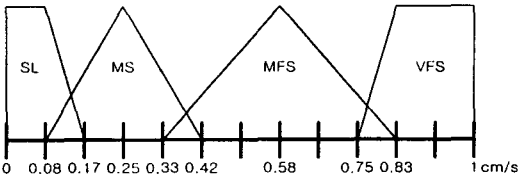


그림 9. 이동로봇의 현재속도 및 출력 속도에 대한 소속함수

fig 9. Membership function of v_L and v_o

<그림10>은 이동로봇과 장애물간의 거리에 대한 소속함수로 이동로봇의 주행방향으로 좌·우 45도 내에 100cm 거리에 있는 물체를 장애물로 인식하도록 하였다.

NE : near , ME : medium FA : far

<그림11>은 이동로봇이 현재 취하고 있는 지향각(θ_1)에 대한 소속함수를 나타내었다.

NB : negative big , ZE : zero

PB : positive big

<그림12>는 이동로봇이 장애물을 인식하고 회피하는 방향각(θ_r)에 대한 소속함수를 나타내었다.

NB , NM , ZE , PM , PB

N : negative , P : positive

M : medium , ZE : zero

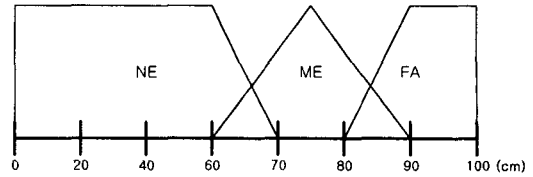


그림 10. 이동로봇과 장애물간의 거리에 대한 소속함수

fig 10. Membership functions of the distance relation between the robot and obstacle

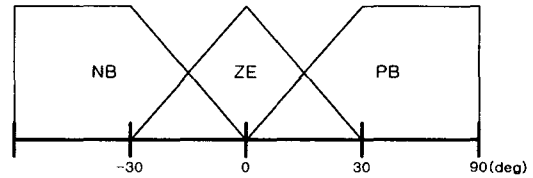


그림 11. 이동로봇의 현재 지향각에 대한 소속함수

fig 11. Membership functions of the robot heading

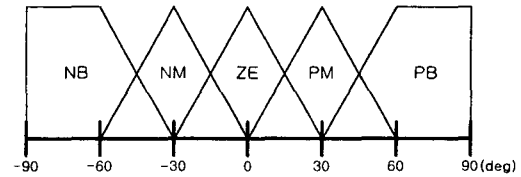


그림 12. 장애물에 대한 이동로봇의 회피 방향각의 소속함수

fig 12. Membership function of the obstacle avoidance angle of the robot

제어규칙의 IF~THEN 형태로 FLC1의 제어규칙은 37개, FLC2의 제어규칙은 22개, FLC3의 제어규칙은 22개로 구성하였다.

퍼지 추론은 Larson 법[4]을 이용, 비퍼지화는 무게 중심법을 사용하였다

표 1. FLC 1

Table 1. FLC 1

Rule	d	θ	L_1, L_2	R_1, R_2	θ_o
Rule 1	-	-	NEAR	-	PS
Rule 2	-	-	-	NEAR	NS
Rule 3	NE	NB	-	-	NB
Rule 4	NE	NM	-	-	NM
Rule 5	NE	NS	-	-	NS

Rule 6	NE	ZE	-	-	ZE
Rule 7	NE	PS	-	-	PS
Rule 8	NE	PM	-	-	PM
Rule 9	NE	PB	-	-	PB
Rule 10	MN	NB	-	-	NB
Rule 11	MN	NM	-	-	NM
Rule 12	MN	NS	-	-	NS
Rule 13	MN	ZE	-	-	ZE
Rule 14	MN	PS	-	-	PS
Rule 15	MN	PB	-	-	PM
Rule 16	MN	PM	-	-	PB
Rule 17	ME	NB	-	-	NB
Rule 18	ME	NM	-	-	NM
Rule 19	ME	NS	-	-	NS
Rule 20	ME	ZE	-	-	ZE
Rule 21	ME	PS	-	-	PS
Rule 22	ME	PM	-	-	PM
Rule 23	ME	PB	-	-	PB
Rule 24	MF	NB	-	-	NB
Rule 25	MF	NM	-	-	NM
Rule 26	MF	NS	-	-	NS
Rule 27	MF	ZE	-	-	ZE
Rule 28	MF	PS	-	-	PS
Rule 29	MF	PM	-	-	PM
Rule 30	MF	PB	-	-	PB
Rule 31	FA	NB	-	-	NB
Rule 32	FA	NM	-	-	NM
Rule 33	FA	NS	-	-	NS
Rule 34	FA	ZE	-	-	ZE
Rule 35	FA	PS	-	-	PS
Rule 36	FA	PM	-	-	PM
Rule 37	FA	PB	-	-	PB

표 2. FLC 2
Table 2. FLC 2

Rule	d	V_L	L_1, L_2	R_1, R_2	V_o
Rule 1	-	-	NEAR	-	SL
Rule 2	-	-	-	NEAR	SL
Rule 3	NE	SL	-	-	SL
Rule 4	NE	MS	-	-	SL
Rule 5	NE	MFS	-	-	MS
Rule 6	NE	VFS	-	-	MFS
Rule 7	MN	SL	-	-	MS
Rule 8	MN	MS	-	-	MS
Rule 9	MN	MFS	-	-	MS
Rule 10	MN	VFS	-	-	MFS

Rule 11	ME	SL	-	-	MS
Rule 12	ME	MS	-	-	MFS
Rule 13	ME	MFS	-	-	MFS
Rule 14	ME	VFS	-	-	MFS
Rule 15	MF	SL	-	-	MS
Rule 16	MF	MS	-	-	MFS
Rule 17	MF	MFS	-	-	MFS
Rule 18	MF	VFS	-	-	VFS
Rule 19	FA	SL	-	-	MS
Rule 20	FA	MS	-	-	MFS
Rule 21	FA	MFS	-	-	VFS
Rule 22	FA	VFS	-	-	VFS

표 3. FLC 3
Table 3. FLC 3

Rule	F_1, F_2	θ_l	l_d	r_d	θ_r
ld < rd					
Rule 1	-	-	NEAR	-	PM
Rule 2	-	-	-	NEAR	NM
Rule 3	NE	NB	-	-	PB
Rule 4	NE	ZE	-	-	PB
Rule 5	NE	PB	-	-	PM
Rule 6	ME	NB	-	-	PB
Rule 7	ME	ZE	-	-	PB
Rule 8	ME	PB	-	-	PM
Rule 9	FA	NB	-	-	PM
Rule 10	FA	ZE	-	-	PM
Rule 11	FA	PB	-	-	PM
ld > rd					
Rule 12	-	-	NEAR	-	PM
Rule 13	-	-	-	NEAR	NM
Rule 14	NE	NB	-	-	NM
Rule 15	NE	ZE	-	-	NB
Rule 16	NE	PB	-	-	NB
Rule 17	ME	NB	-	-	NM
Rule 18	ME	ZE	-	-	NB
Rule 19	ME	PB	-	-	NB
Rule 20	FA	NB	-	-	NM
Rule 21	FA	ZE	-	-	NM
Rule 22	FA	PB	-	-	NM

IV. 시뮬레이션

1. 시뮬레이션에 사용한 원격제어 이동로봇의 형태

- i) 드라이브 방식 : 두 개의 분리된 바퀴를 따로 구동하는 방식
- ii) 이동로봇의 무게 중심을 위한 castor.
- iii) 초음파센서가 360도 회전하면서 지도 정보를 위한 거리정보 수집하고 샘플링 시간은 2.38ms.

- iv) 좌·우측 벽면과의 거리측정은 좌·우측에 45도 간격으로 4개의 레이저 센서로 측정
- v) 전방 장애물 검출을 위해 2개의 레이저센서로 측정, 샘플링 시간은 90ns
- v) 초음파 센서[2]의 회전은 스텝모터로 구동.
- vi) 이동 로봇의 지향각은 지자기센서로 측정
- vii) 드라이브 모터는 스텝 모터(1.8 °/step)

2. 시뮬레이션 환경

- i) 바뀌는 미끄러짐이 없는, 완전 회전을 전제로 한다.[5]
- ii) 이동 로봇과 좌·우측 벽과의 거리는 레이저센서가 수집한 거리정보 중 가장 짧은 좌·우측 거리를 선택한다.
- iii) 이동로봇이 정지상태일 때 초음파 센서로 지도정보를 수집하고 주행중일 때는 좌·우측 벽면과의 거리정보 및 장애물 정보만 수집한다.

3. 시뮬레이션 결과

i) 시뮬레이션 1·2

시뮬레이션 1에서 원격이동로봇이 4개의 주행 명령으로 주행한다면 직선점선과 같이 3개의 경유점을 통과해서 목적지에 도착하게 되지만 제안된 주행알고리즘으로 시뮬레이션 한 결과, 표1에서와 같이 2개의 경유점 만을 지나고 총 주행거리를 비교해보면 퍼지 알고리즘을 적용한 경우가 더 짧은 거리로 주행했음을 볼 수 있다.

시뮬레이션 2에서는 원격이동 로봇이 4개의 주행 명령으로 주행한다면 3개의 경유점을 통과해서 도착하게 되지만 제안된 주행 알고리즘으로 시뮬레이션 한 결과 표2에서와 같이 경유점 없이 도착하고 총 주행거리도 퍼지 알고리즘을 적용한 경우가 더 짧은 거리로 주행했다. 그리고, 제안한 알고리즘대로 이동 로봇이 좌·우회전하는 구간에서 벽과 부딪히지 않게 일정거리를 유지하면서 부드럽게 속도를 줄이면서 통과했다. <그림15><그림16>에서 직선으로 된 점선이 기존 이동 명령으로 주행하였을 때이고 곡선주행의 점선이 퍼지 알고리즘을 이용했을 때와 그 경유점을 나타낸다.

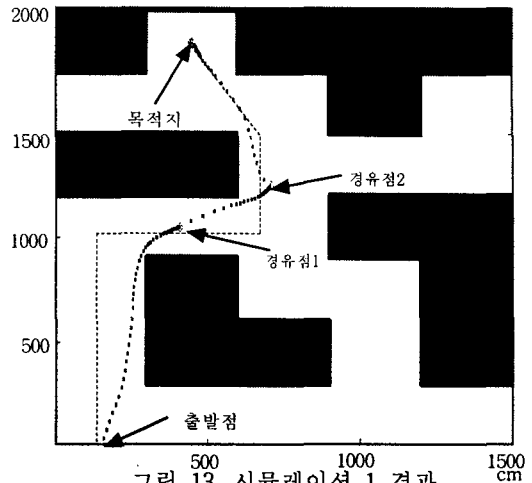


그림 13. 시뮬레이션 1 결과
fig 13. Simulation 1 result

표 6 . 시뮬레이션 1 데이터
Table 1. Simulation 1 data

출발점	경유점1	경유점2	목적지	초기지향각
(150,0)	(410,1050)	(710,1250)	(450,1950)	0°
최대속도구간	(252,581)~(258,654)		36.49cm/s	
지향각 변화	-59° ~ 14°			
총 주행거리	퍼지알고리즘을 수행시			2291cm
	기존명령으로 수행시			2640cm

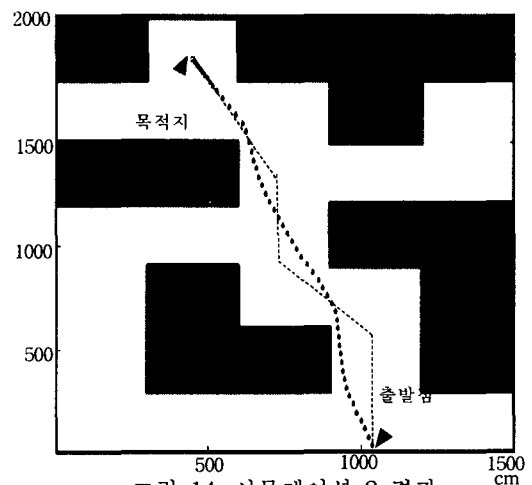


그림 14. 시뮬레이션 2 결과
fig 14. Simulation 2 result

표 7. 시뮬레이션 2 데이터

Table 2. Simulation 2 data

출발점	경유점	목적지	초기지향각	지향각변화
(1050,0)	없음	(450,1900)	0°	-18° ~ 5°
최대속도구간	(781,1011)~(730,1146)			36.22cm/s
	(1015,108)~(971,246)			
총 주행거리	퍼지알고리즘 수행시	2002.8cm		
	기본명령으로 수행시	2100.8cm		

ii) 시뮬레이션 3

시뮬레이션3은 시뮬레이션1의 주행경로 (258.715)와 좌표(491.1104)에 지도에 없는 장애물1과 장애물2를 놓았을 때 장애물을 회피하는 시뮬레이션이다.

장애물1은 가로,세로 40cm 크기이고 장애물2는 가로90cm 세로40cm 크기이다

<그림15>에서 장애물을 감지하면 좌측 또는 우측으로 장애물을 회피할 수 있는 방향을 판단, 회피방향으로 회전하여 목적지에 도달하였다.

FLC3이 충돌회피각도 계산 횟수가 평균 3번 이내로 속도계산을 위한 새로운 FLC 보다는 시뮬레이션 1과 2에서의 평균 곡선주행 속도인 20cm/s로 주행하도록 하였다

V. 결 론

본 논문은 퍼지 논리 제어기를 이용해서 원격제어 이동로봇이 출발점에서 목적지까지 경유점을 줄이면서 장애물이 있을 경우 장애물 회피주행을 안전하고 부드럽게하기 위한 방법을 제안하고 시뮬레이션을 통해 유용성을 검증하였다.

그리고, 주행 알고리즘 개선을 위해 좌·우거리와 전방 장애물 검출을 위한 레이저 센서를 가정하고 알고리즘을 개발하였다. 따라서, 원격제어 이동로봇의 구조를 개선해서 FLC 주행을 적용하는 실험을 수행중에 있다.

접수일자 : 2000. 9. 7. 수정완료 : 2000. 10. 18

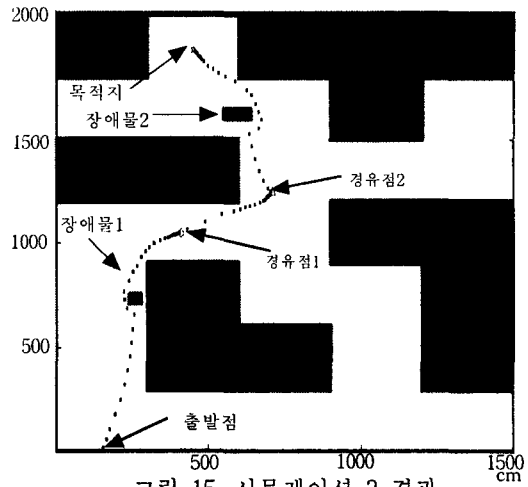


그림 15. 시뮬레이션 3 결과

fig 15. Simulation 3 result

표 8. 시뮬레이션 3 데이터

Table 3. Simulation 3 data

장애물 위치	장애물1	(280,710)~(240,750)	
	장애물2	(640,1640)~(550,1600)	
장애물1 회피	좌측거리	지향각	장애물거리
	우측거리	회피각도	이동속도
장애물2 회피	좌측거리	지향각	장애물거리
	우측거리	회피각도	이동속도

VI. 참고 문헌

- [1] 이상원, 이상식, "군사적 활용을 위한 원격 제어 이동 로봇에 대한 연구", 한국 군사 과학 기술 학회 추계 학술 대회 논문집 2권, pp.397~402, 1999
- [2] Polaroid ultrasonic ranging unit data book, pp.1~5, 1999
- [3] M.Sugeno and M.Nishida "Fuzzy Control of Model Car", 1985 North-Holland. Retyped with written permission from Fuzzy Set and Systems, 16 (1985) pp.103~113
- [4] Lefteri H. Tsoukalas, Robert E. Uhrig, "Fuzzy and Neural Approaches in Engineering ", John Wiley & Sons INC , pp. 14 5~188, 1996

[5] 박찬규, 최정원, 권순학, 이석규 “퍼지-뉴럴을 이용한 이동 로봇의 장애물 충돌 회피”, 제어·자동화·시스템공학 논문지 제5권 제8호 pp. 990~994, 1998



정지봉(Ji bong Jung)
准會員
1999년 경남대학교 전자공학과 졸업(공학사)
1999-현재 경남대학교 메카트로닉스과 석사과정
관심분야 : 시스템 지능제어, 로보틱스



이상식(Sang-sik Lee)
正會員
1978년 해군사관학교 전자과 (이학사)
1984년 서울대학교 전자과 (공학사)
1987년 미국해군대학원 (N.P.G.S) (공학석사)

1992년 영국 웨일즈 카디프대학 (공학박사)
1997-1998년 군사과학대학원 해양공학 주임교수
1998-1999년 해군사관학교 공학처장
1999-현재 해군사관학교 전자과 부교수
관심분야 로보틱스 및 응용, 지능제어시스템, 회로 및 시스템



신위재(Wee-jae Shin)
正會員
1975년 동아대학교 전자공학과 졸업
1979년 동대학원 전자공학과 (공학석사)
1989년 동대학원 전자공학과 (공학박사)

1996-1997년 대한전자공학회 부산경남 지부장
1998-1999년 대한전자공학회 학술위원
1998년 일본 Kyushu 대학원 객원교수
2000년-현재 한국신호처리·시스템학회 부회장
1983년-현재 경남대학교 전기전자공학부 교수
관심분야 : 시스템지능제어 및 자동화, 신호처리