

工學碩士學位論文

이동로봇의 인터넷 기반 원격제어 시
시간지연에 따른 경로 보상 제어

2001年 11月

昌原大學校 大學院

制御計測工學科 Intelligent Mechatronics 專攻

劉 鳳 洙

工學碩士學位論文

이동로봇의 인터넷 기반 원격제어 시
시간지연에 따른 경로 보상 제어

Path Compensation Control for Remote
Control of Internet Based Mobile Robot with
Time Delay

指導教授 趙重宣

이 論文을 工學碩士學位論文으로 提出함

2001年 11月 日

昌原大學校 大學院

制御計測工學科 Intelligent Mechatronics 專攻

劉 鳳 洙

劉鳳洙의 碩士學位 論文을 認准함.

審 查 委 員 長 조철우 ①

審 查 委 員 김재형 ①

審 查 委 員 조중선 ①

2001年 11月 日

昌原大學校 大學院

목 차

제1장 서론	1
제2장 이동로봇의 구성	3
2.1 이동로봇의 개요	3
2.2 이동로봇의 기구학적 모델	4
2.3 이동로봇의 전체시스템 구조	7
2.3.1 구동부	8
2.3.1.1 제어기(80C196KC)	12
2.3.1.2 컨버터(Converter)	14
2.3.1.3 엔코더 카운터	16
2.3.2 센서부	16
2.3.2.1 초음파 센서(Ultrasonic Sensor)	16
2.3.2.2 스테레오 비전(Stereo Vision)	19
제3장 인터넷 기반 이동로봇 시스템에서의 시간지연	22
3.1 인터넷 기반 이동로봇 시스템	22
3.2 시간지연의 분류	23
제4장 인터넷 기반 이동로봇의 시간지연 보상 알고리즘	26
4.1 인터넷 기반 이동로봇의 시간지연 보상의 개념	26
4.2 퍼지논리를 이용한 인터넷 기반 이동로봇의 시간지연 보상	27
4.2.1 A구간 경로 생성 알고리즘	29

4.2.2 B구간 경로 생성 알고리즘	30
4.2.3 C구간 경로 생성 알고리즘	33
4.2.4 시간오차 보상 알고리즘	34
제5장 시뮬레이션	38
5.1 시뮬레이션을 위한 파라미터	38
5.2 시간지연 발생 시 적절한 보상이 없는 경우	39
5.3 시간지연에 대한 적절한 보상이 적용된 경우	40
5.4 퍼지논리를 이용하여 시간지연을 보상한 경우	41
제6장 결론	43
참 고 문 헌	44
Abstract	47

표 목 차

표 1. 구동부 사양	10
표 2. 제어신호	12
표 3. P_{rf} 생성 알고리즘의 Rule Table	32
표 4. 시뮬레이션에 사용된 시간테이블	38
표 5. 시간지연 보상에 따른 성능지수 비교	42

그림 목 차

그림 1. 이동로봇의 구조와 기구학적 모델	4
그림 2. 제작된 이동로봇의 실제 모습	7
그림 3. 이동로봇의 전체구조	8
그림 4. 이동로봇의 하드웨어 구성도	9
그림 5. 구동부의 구성도	9
그림 6. 모터의 속도제어를 위한 구성도	11
그림 7. 실제 제작된 모터 드라이브	11
그림 8. PWM Duty rate	13
그림 9. 실제 80C196KC Board	14
그림 10. H-Bridge 회로	15
그림 11. 컨버터(Converter)	15
그림 12. 엔코더 카운터	16
그림 13. 초음파 센서 구동부 회로	17
그림 14. 초음파 센서 보드	18
그림 15. 초음파 센서 타이밍 다이어그램	18
그림 16. 스테레오 비전	20
그림 17. 프레임 그레이버	20
그림 18. 인터넷 기반 이동로봇 시스템의 전체 구조	22
그림 19. 이동로봇의 경로오차와 시간오차	24
그림 20. 인터넷 기반에서의 시간지연 정의	24
그림 21. 인터넷 기반 이동로봇의 경로 및 시간보상 전체 구조	26
그림 22. 시간지연 발생시 경로오차에 대한 상황	27

그림 23. 제안된 이동로봇의 주행 경로	28
그림 24. 퍼지논리를 이용한 이동로봇의 경로 보상	29
그림 25. 이동로봇의 시간지연 동안 이동한 거리에 대한 소속함수	31
그림 26. 계획된 경로와 실제경로 사이의 각도에 따른 소속함수	31
그림 27. 에 대한 소속함수	31
그림 28. 에서 까지 생성되는 축 경로좌표에 대한 소속함수	33
그림 29. 이동로봇의 속도 프로파일	35
그림 30. 이동로봇의 가감속 시간에 대한 소속함수	36
그림 31. 시뮬레이션에 사용된 시간지연	39
그림 32. 시간지연 발생 시 적절한 보상이 없는 경우	39
그림 33. 시간지연에 따른 적절한 보상이 적용된 경우	40
그림 34. 퍼지논리를 이용하여 시간지연 보상을 한 경우	41
그림 35. 시뮬레이션에 사용된 속도 프로파일	42

제1장 서론

최근 고도화된 정보사회에서 인터넷은 다양한 분야에서 적용사례를 보이고 있다. 이러한 현상은 로봇틱스 분야에서도 마찬가지로 인터넷 기반 원격로봇 시스템은 많은 연구자들이 관심을 가지고 있으며, 인터넷을 기반으로 한 다양한 형태의 로봇이 개발 및 연구되고 있다. Berkeley 대학의 Goldberg는 Mucury project[1]에서 Standard Web Browser를 통해서 Robot arm을 제어하였고, 이후에 Telegarden project[2]에서 흙 속에 묻힌 물건을 찾거나 정원에 물을 주는 간단한 로봇 시스템을 개발하였다. Western Australia 대학의 Telerobot 연구[3]는 인터넷을 통하여 산업용 ASEA IRB6 Robot arm을 이용한 블록 쌓기 등을 하였다. 또한, Schulz과 Burgard[4]등은 이동로봇 Minerva를 개발하였는데, 이 이동로봇은 박물관에서 사람을 안내하는 작업을 성공적으로 수행하였다.

한편 현재의 인터넷기반 로봇기술은 전용통신망이 아닌, 범용통신망을 이용하고 있다. 따라서 원격 조작자 또는 조작컴퓨터와 이동로봇간의 인터넷을 통한 통신 또는 제어는 갑작스런 통신 단절이나 낮은 대역폭, 또는 불특정 다수의 동시접속으로 인한 시간지연 등으로 이동로봇의 오동작의 원인이 된다. 이러한 문제를 해결하기 위하여 Riko[5]등은 원격의 조작자가 인터넷을 통하여 Laboratory robotic system에 접근하여 Manipulator를 제어 시 V-Collide library를 사용하여 Simulator의 데이터 계산 시간을 줄이고 시간지연에 의해서 생기는 Manipulator와 그에 관계된 작업공간상의 물체와의 충돌을 방지하였다. Imad[6]는 이동로봇이 시간지연에 의해 일정시간 내에 명령을 받지 못했다면 장애물과의 거리를 검출하고 만약 장애물과의 거리가 미리 정의되어있는 위험 거리보다 가까우면 로봇을 정지시키는 Event based control을 통하여 Remote

control system의 안정성을 높였고, Ohba[7]는 시간지연이 있는 Network를 사용한 원격 로봇의 협력을 Multi Operator Multi Robot (MOMR) Teleoperation을 통하여 연구하였다. 이때 MOMR 시스템을 수행하는데 있어서 몇 가지의 구속을 두고, 이 구속 하에서 시간지연이 MOMR 시스템에 미치는 영향을 설명하고, MOMR 시스템에서 다른 로봇과의 충돌을 방지하기 위하여 Virtual thickness modification 방법을 제안하였다.

그러나 최근 인터넷관련 기술의 발달로 인해 시간지연이 짧아지고, 컴퓨터 하드웨어 및 로봇기술의 발달로 인해 로봇의 속도가 고속화되었기 때문에 시간지연에 대한 보다 적극적인 대응이 필요하게 되었다. 따라서 본 논문에서는 이러한 여러 가지의 연구들 중 인터넷을 통한 이동 로봇의 원격제어 시 이동로봇에 전달되는 명령에 시간정보를 덧붙임으로써 시간지연을 계산하고, 계산된 시간지연에 따라 통신시간지연을 두 가지 형태로 정의한 후, 퍼지 논리를 사용하여 시간지연에 따른 주행경로 오차 및 시간오차를 보상하는 방법을 제안한다. 그리고 제안된 방법을 시뮬레이션을 통하여 검증하고자 한다.

본 논문은 다음과 같이 구성되었다. 제2장에서는 제작된 이동로봇의 전체시스템 구조와 기본 하드웨어를 부분별로 나누어 설명한다. 제3장에서는 인터넷 기반 이동로봇 시스템에서의 시간지연을, 제4장에서는 인터넷 기반 이동로봇 시스템의 전체구조와 본 논문에서 제안하는 시간지연 보상 알고리즘에 대해서 설명한다. 제5장에서는 시뮬레이션 결과를, 마지막 제6장에서는 결론을 맺는다.

제2장 이동로봇의 구성

2.1 이동로봇의 개요

이동로봇(mobile robot)은 인간의 조작없이도 주어진 임무를 자율적으로 수행할 수 있는 시스템을 말한다. 이러한 이동로봇에 대한 연구는 정형화된 작업 환경을 가정하는 매니플레이터(manipulator)와는 달리 많은 불확실성이 포함된 실제 환경의 가정하에서 이루어져야만 한다. 특히 미지의 환경에서 이동로봇이 예기치 못하던 장애물을 안전하게 회피하여 목표점까지 도달하도록 하는 것은 매우 중요한 연구과제 중의 하나이다. 이를 위해서는 미지의 주변 환경을 이동로봇 스스로가 인식할 수 있는 장치 및 이를 해석할 수 있는 방법이 필요하며 또한 현재의 자기 위치에서부터 목표점까지 안전한 경로를 실시간으로 생성시킬 수 있는 알고리즘이 필수적으로 요구된다.

이동로봇은 스스로 목표점 또는 계획된 경로를 이동하거나 주어진 작업을 수행할 수 있는 구동부, 상황의 판단 및 주어진 임무를 실시간으로 계산하고 처리할 수 있는 연산부, 그리고 미지의 주위환경을 인식하여 장애물이나 물체를 회피하여 목표점까지 도달하도록 하는 센서부 등으로 이루어진다. 주변환경을 인식하기 위한 장치로는 초음파 센서(Ultrasonic Sensor), 스테레오 비전(Stereo Vision), 적외선 센서(Infrared Sensor)등이 있는데 스테레오 비전 시스템의 경우 정확한 정보를 제공할 수 있지만 가격이 고가인 단점이 있고, 그 데이터를 처리하는데 많은 시간이 소요되어 실시간 제어가 불가능한 경우가 많으며, 속도 개선을 위해 별도의 처리장치를 필요한 경우가 있다. 초음파 센서와 적외선 센서는 가격

이 저렴하고 실시간 계산이 가능하다는 장점이 있지만, 정확도가 떨어진다는 단점이 있다. 지금까지는 단일 센서를 이용한 연구가 주류를 이루어 왔지만 최근 들어 컴퓨터 하드웨어와 주변기기의 발달로 인한 처리속도의 증가로 상황에 따라 적절한 센서를 결합하여 사용하는 연구가 활발하게 진행되고 있다.

2.2 이동로봇의 기구학적 모델

이동로봇의 동작은 위치이동을 기본으로 이루어진다. 따라서 안정된 위치이동의 보장은 이동로봇의 제어 시스템이 기본적으로 갖추어야 할 기능이다. 본 절에서는 이동로봇의 모델과 제한 조건에 대해 설명한다.

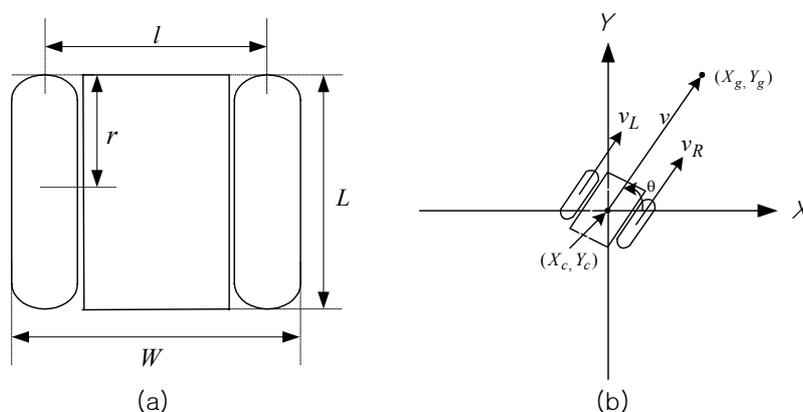


그림 1. 이동로봇의 구조와 기구학적 모델

이동로봇의 구조는 그림 1의 (a)와 같다. l 은 두 바퀴 사이의 거리가

다. r 은 바퀴의 반지름, W 와 L 은 각각 로봇의 가로와 세로길이를 나타낸다. 이동로봇의 바퀴의 회전축은 바닥의 수직성분과 직교하며 바퀴와 바닥사이의 접촉면에서 Pure rolling과 Non-slipping의 조건을 만족한다. Pure rolling은 바퀴 표면의 점들이 연속적으로 바닥 표면의 점들과 일대일 접촉을 할 때를 말하며, Non-slipping은 이동로봇이 나아가는 방향의 수직 성분으로 미끄러지지 않는다는 것을 말하며, 이동로봇은 이동 중 옆으로 미끄러지지 않는다는 것을 의미한다.

이동로봇의 기구학적인 모델은 그림 1의 (b)에 의해 설명될 수 있다. 2차원 평면으로 구성된 전역 좌표계(Global Coordinate System)에서 이동로봇의 상태를 식 (2.1)과 같이 위치와 방향을 갖는 벡터로 표현한다.

$$P = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad (2.1)$$

일반적으로 이동로봇의 운동은 두 개의 제어입력인 선속도와 각속도에 의해 제어되는 시스템으로 모델링 된다. 이동로봇의 왼쪽 바퀴와 오른쪽 바퀴의 각속도를 각각 w_L , w_R , 왼쪽 바퀴와 오른쪽 바퀴의 지면과의 접촉점의 선속도를 각각 v_L , v_R , 그리고 이동로봇의 제어 입력으로 인가되는 이동로봇의 선속도와 각속도를 각각 v , w 라고 하면, 각 변수들은 식 (2.2)와 식 (2.3)과 같은 관계를 갖는다.

$$\begin{bmatrix} v_R \\ v_L \end{bmatrix} = r \begin{bmatrix} w_R \\ w_L \end{bmatrix} \quad (2.2)$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{l} & \frac{1}{l} \end{bmatrix} \begin{bmatrix} v_R \\ v_L \end{bmatrix} \quad (2.3)$$

또한 로봇의 자세 벡터 P 와 이동로봇의 제어 입력 u 는 자코비안 행렬 J 에 의해 식 (2.4)와 같이 표현된다.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = J \cdot u = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (2.4)$$

식 (2.4)와 같은 시스템은 논홀로노믹(non-holonomic) 제약을 가진다. 즉 2구동륜의 이동로봇과 같이 구조적인 자유도가 2개인 시스템은 평면 상에서의 좌표요소인 (x, y, θ) 3자유도를 원하는 대로 제어할 수 있는 스무드한 시불변의 피드백 제어 규칙이 존재하지 않는다는 것이다. 따라서 현재 이동로봇의 자세에서 원하는 자세로 제어할 때 제한조건이 발생한다. 바닥과 바퀴의 접촉면에서 바퀴면에 수직인 방향의 속도성분은 0이라는 논홀로노믹 조건을 이용하면 식 (2.5)와 같은 논홀로노믹 제한조건을 구할 수 있다.

$$H \cdot \dot{p} = (\sin \theta - \cos \theta) \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \quad (2.5)$$

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0 \quad (2.6)$$

여기서, H 는 바퀴면에 수직인 단위 벡터이다.

제한 조건은 식 (2.6)과 같이 되며, 이것은

$$\tan \theta = \frac{\dot{x}}{\dot{y}} \quad (2.7)$$

식 (2.7)과 같이 정리될 수 있다. 이것은 이동로봇의 순간 진행 방향은 이동로봇이 향하고 있는 각도 θ 와 같아야 한다는 의미이다.

2.3 이동로봇의 전체시스템 구조

실제 제작된 이동로봇의 실제 모습은 그림 2와 같다.

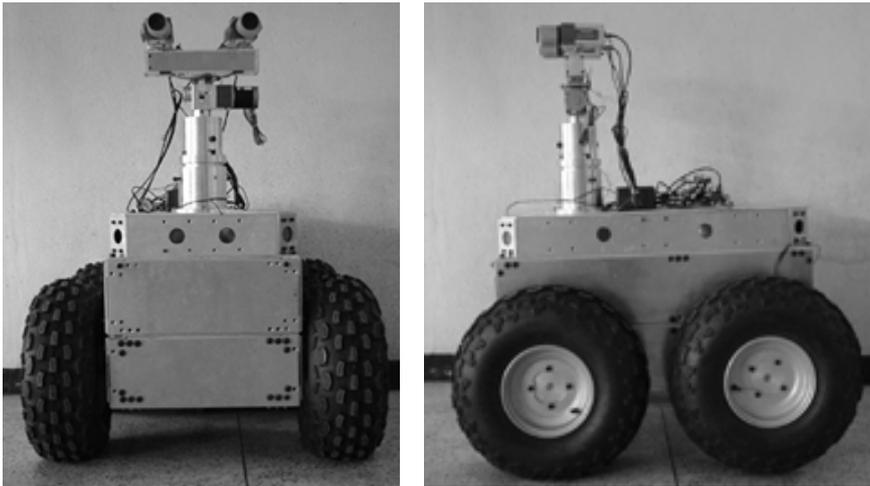


그림 2. 제작된 이동로봇의 실제 모습

개발된 이동로봇은 두 개의 DC Servo Motor로 구동할 수 있는 네 개의 바퀴로 움직이며, 열 두개의 초음파 센서를 모듈로 장착을 하고 있다.

또한 두 대의 CCD 카메라를 사용하여 스테레오 비전으로 구성하였고, 프로세서간의 통신을 위하여 CAN 버스를 사용하였다. 주 컴퓨터와 모터 구동 프로세서간의 통신을 위하여 시리얼 통신과 RF 모듈을 사용하였다. 또한 네트워크 통신을 위하여 무선랜을 사용한다. 이동로봇을 구동하는 전원으로 DC 12V 배터리를 사용하여 DC 12V, 5V, 75V로 DC/DC 변환하여 센서 보드, 모터 드라이버, 구동 모터 등에 필요한 전원을 공급하고, AC 220V로 DC/AC 변환하여 주 컴퓨터의 전원으로 사용한다. 이동로봇의 크기는 가로, 세로, 높이가 각각 100cm, 110cm, 120cm이며, 본체의 재질은 알루미늄으로 구성되어 있다. 스테레오 비전은 상하좌우로 움직이는데 사용되는 DC 모터 두 개와 카메라를 좌우로 움직이는 스테핑 모터 두 개로 구성된다. 그림 3은 이동로봇의 전체구조이다.

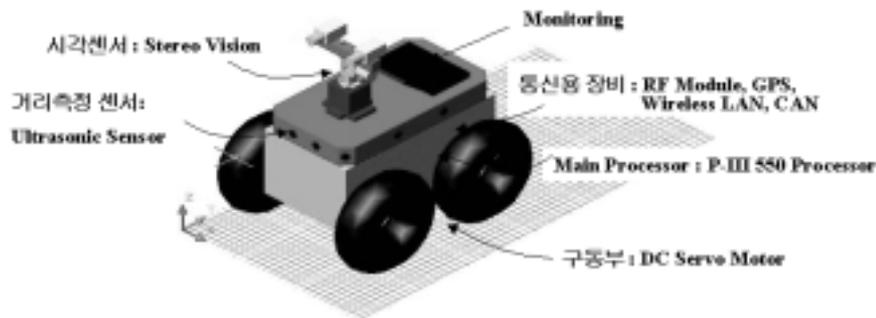


그림 3. 이동로봇의 전체구조

2.3.1 구동부

이동로봇은 네 개의 바퀴에 의해 움직이며, 그림 4와 같이 모터를 구동시키기 위한 모터 드라이브와 DC 서보 모터 2개, 그리고 감속기와 벨트 및 풀리, 엔코더 등으로 구성된다.

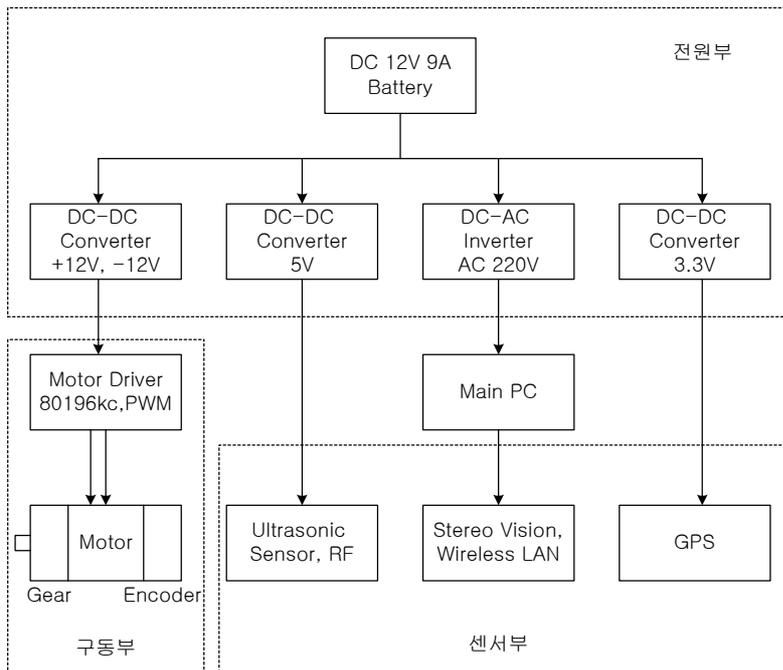


그림 4. 이동로봇의 하드웨어 구성도

모터 드라이브는 속도제어를 사용하였다. DC 서보 모터의 속도 제어를 위한 전체 시스템 구조는 그림 5와 같다. 서버(server) PC에서 속도 명령이 내려지면 제어기(Controller)는 엔코더 카운터(Encoder Counter)로부터 모터의 속도 값을 입력받아 제어 명령을 출력한다. 컨버터는 제어 출력

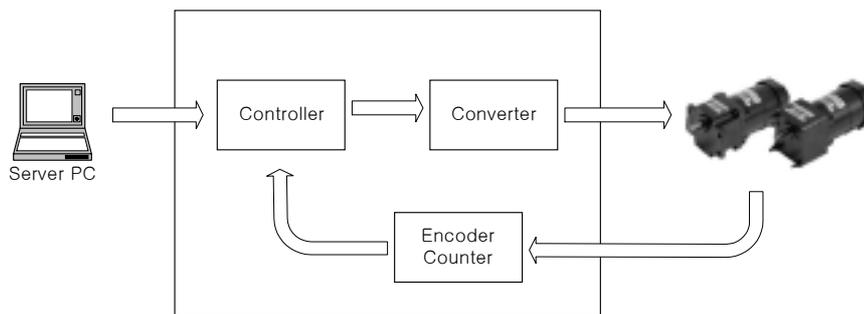


그림 5. 구동부의 구성도

값을 전압으로 변환하여 모터를 구동하며 제어기의 샘플링 시간(Sampling time)은 10ms 이다.

속도제어를 위해 사용된 DC 모터의 구동 방식은 트랜지스터에 의한 PWM방식이 사용되었다. 모터에 장착된 인크리멘탈 엔코더의 엔코더 값을 계수 하기 위해서 엔코더 전용 카운터 HCTL-2016을 사용하였고, 제어기는 MCS-96 계열의 80C196KC를 사용하였다.

또한 구동부 사양은 표 1 과 같다.

모터 출력	110W
입력 전압	48V
Encoder	4000 Puls/Res
기어 비	1:15 (기어:모터)
폴리 비	1:3 (바퀴:기어)
바퀴 지름	500 mm
최대 속도	1m/s
속도 Resolution	256단계
샘플링 시간	10m/s
Controller	80C196KC
제어 방식	PI 제어
모터 구동방식	PWM
PWM Switching Time	15.625KHz
사용 언어	어셈블러

표 1. 구동부 사양

목표속도 32rps는 이동로봇이 1m/s로 이동하기 위한 모터의 회전속도이다. 모터의 정격 회전속도가 50rps이므로 목표속도 32rps를 내기 위해서 모터의 전원을 48V로 하였다.

그림 6은 모터의 속도제어를 위한 블록 다이어그램이고, 그림 7은 실제 제작된 모터 드라이버이다.

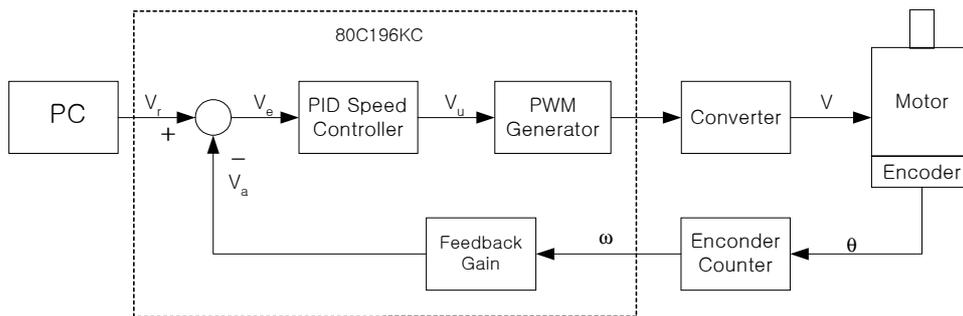


그림 6. 모터의 속도제어를 위한 구성도

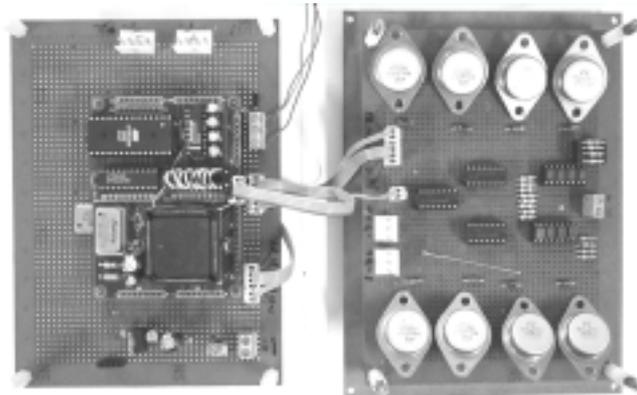


그림 7. 실제 제작된 모터 드라이버

모터 드라이버는 크게 제어기(80C196KC), 컨버터, 엔코더 카운터로 구성되어 있다.

2.3.1.1 제어기(80C196KC)

제어기로 사용된 80C196KC는 4개의 포트를 가지고 있다. 본 이동로봇 시스템에서는 모터 드라이버, 엔코더 카운터, 시리얼 통신을 제어하기 위하여 포트 1, 포트 2를 사용한다. 표 2는 각 포트의 제어 신호들이다.

Port	제어신호
Port 1.0	Encoder Reset
Port 1.1	Motor 1 Phase
Port 1.2	Motor 1 Enable
Port 1.3	Motor 1 PWM
Port 1.4	Motor 2 PWM
Port 1.5	Motor 2 Phase
Port 1.6	Motor 2 Enable
Port 2.0	TxD
Port 2.1	RxD

표 2. 제어신호

제어기는 엔코더 카운터로부터 엔코더 값을 받는다. 계수된 엔코더 값을 받은 후 제어기는 엔코더 카운터에 리셋(Reset) 신호를 보내어 엔코더 값을 리셋 시킨다. 이렇게 되면 θ 값을 샘플링 시간(10 ms)으로 미분하는 결과가 되어 θ 는 ω_a 가 되어 피드백 된다.

그리고 모터를 제어하기 위한 출력신호는 PWM 값으로 보내어 진다. DC 모터의 속도를 연속적으로 바꾸려면 기본적으로는 DC 모터에 가하는 전압을 바꾸면 속도는 변한다. 모터의 코일에 흐르는 전류와 속도가 정비례하기 때문에 단순히 모터의 구동전압을 변화시키면 속도를 가변 할 수 있

다. PWM 방식은 트랜지스터의 활성영역(On-Off)만 사용하므로 트랜지스터에서 소비하는 전력이 없다. 또한 디지털 회로와의 호환이 뛰어나기 때문에 모터의 속도 제어뿐만 아니라 전압 제어를 필요로 하는 대부분의 회로에 응용된다.

80C196KC는 3개의 PWM 출력을 제공하며, 각각의 PWM 파형은 주파수가 15.625 KHz이다. PWM 듀티(Duty) 비는 레지스터의 값에 따라 프로그램적으로 조절할 수 있다. 레지스터는 8비트이며 결과적으로 256단계의 분해능을 가진다. 그림 8은 80C196KC의 PWM 레지스터 값에 따른 듀티 비의 변화를 나타낸 것이며, 그림 9는 실제 사용된 80C196KC 보드이다.

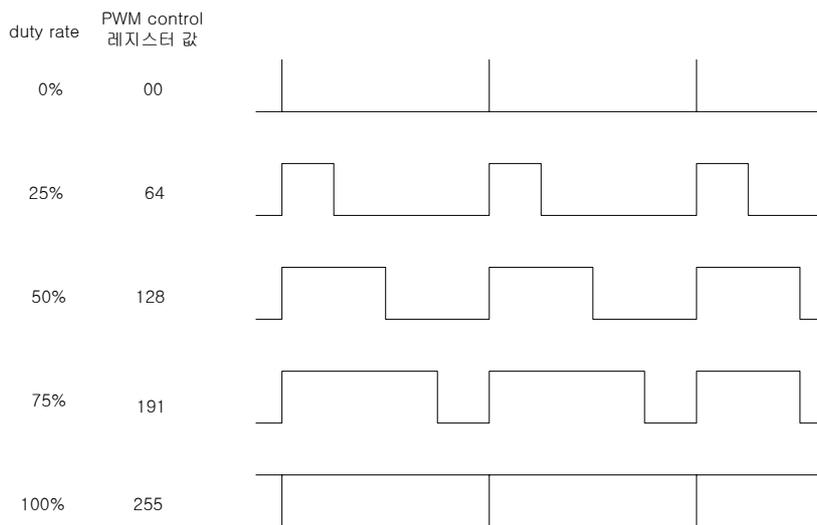


그림 8. PWM Duty rate

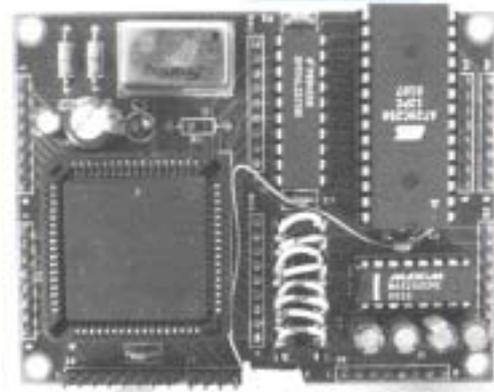


그림 9. 실제 80C196KC Board

2.3.1.2 컨버터(Converter)

컨버터는 제어기로부터 나오는 PWM 신호의 평균 전압 값에 따라 모터의 회전 속도를 제어하게 되는데, 이때 제어기의 5V PWM 전압레벨을 모터의 48V 전압레벨로 증폭시켜주는 역할을 한다. 모터의 전원과 나머지 보드의 전원을 포토 커플러를 사용하여 분리하였다. 이것은 모터에서 발생하는 노이즈가 나머지 로직 회로에 영향을 미치지 못하도록 하기 위한 방법이다.

또한 모터의 방향을 제어하기 위하여 H-bridge 회로를 사용하였다. 그림 10에서 Q1, Q4를 ON하고 Q2, Q4를 OFF하면 모터의 1번에서 2번으로 전류가 흐르고 Q2, Q4를 ON하고 Q1, Q3를 OFF하면 모터의 2번에서 1번으로 전류가 흐른다. H-bridge 회로는 중소형 모터에 가장 많이 적용되는 회로이다. 그림 11은 실제 제작된 Converter이다.

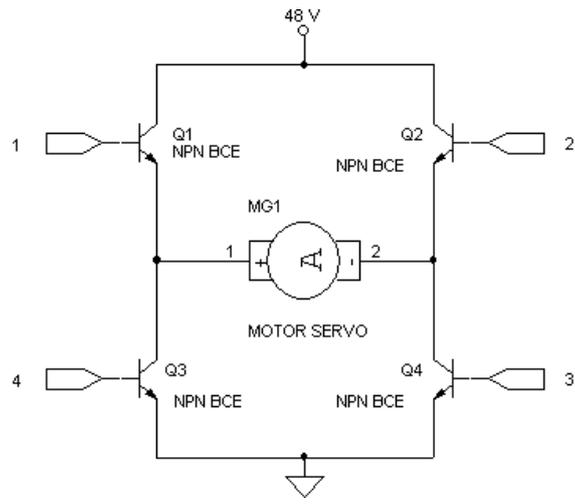


그림 10. H-Bridge 회로

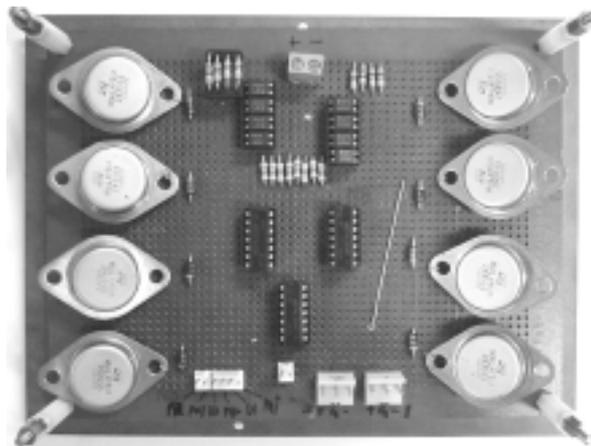


그림 11. 컨버터(Converter)

2.3.1.3 엔코더 카운터

엔코더 카운터는 16 bit 엔코더 카운터 소자(HCTL-2016)와 수정 발진기를 사용하여 간단하게 구성할 수 있다. 그리고 엔코더 값이 80C196KC의 데이터 버스로 전송되게 설계하기 위하여 74LS138을 이용한 디코더회로가 추가되었다. 그림 12는 본 이동로봇 시스템에서 개발된 엔코더 카운터이다.

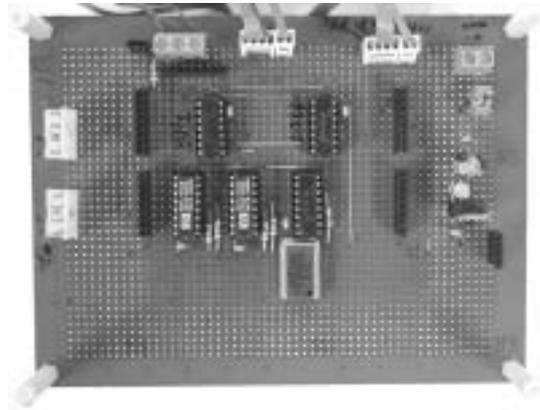


그림 12. 엔코더 카운터

2.3.2 센서부

2.3.2.1 초음파 센서(Ultrasonic Sensor)

이동로봇이 주행 중에 장애물 회피나 거리측정 또는 주어진 명령의 수행 여부 등을 판단하기 위해서는 이동로봇과 주위 물체 사이의 거리정보가 필요하다. 이동로봇의 주위환경을 인식할 수 있는 센서는 여러 가지가 있지만 그중 초음파 센서는 초음파 빔을 발사한 후 물체에 반사되어 돌아온 시간을 측정하여 장애물과의 거리를 측정하는 센서이다. 이때

반사되어 돌아오는 초음파는 공기중의 퍼짐 효과와 소리의 공기 중 희석 효과로 시간이 지남에 따라 그 크기가 감소한다. 따라서 시간의 흐름에 따라 이득이 커지는 증폭기를 사용하여 공기 중에서 약해진 초음파의 크기를 보상해 준다. 따라서 시간의 흐름에 따른 초음파의 크기에 관계없이 돌아오는 초음파를 감지할 수 있다. 제작된 이동로봇시스템에 사용된 거리 측정용 초음파 센서로는 폴라로이드 6500 시리즈 초음파 센서를 사용하였다. 이는 송신과 수신이 하나의 센서에서 이루어지고 구동주파수는 50kHz이며 지향성은 15°정도의 특성을 갖는다. 또한 지면으로부터 60cm의 높이에 12개의 초음파 센서 모듈을 사용하여 이동로봇 주위의 장애물을 식별하거나 거리정보 등을 획득하였다. 데이터의 통신을 위해서 USB 포트를 사용하였다.

그림 13은 초음파 센서를 구동하기 위한 회로이고, 초음파 센서 보드는 그림 14와 같다.

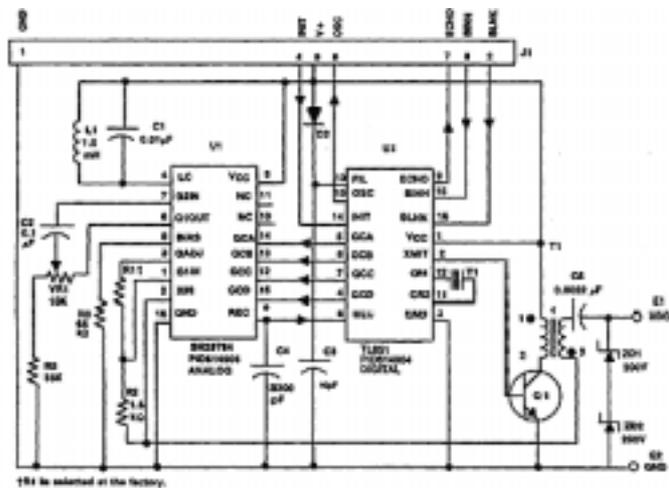


그림 13. 초음파 센서 구동부 회로



그림 14. 초음파 센서 보드

Acroname사의 폴라로이드 6500센서의 유효거리는 12cm ~ 10m 정도이고 동작 주파수는 50kHz를 사용한다. 이 회로의 타이밍도는 그림 15와 같다.

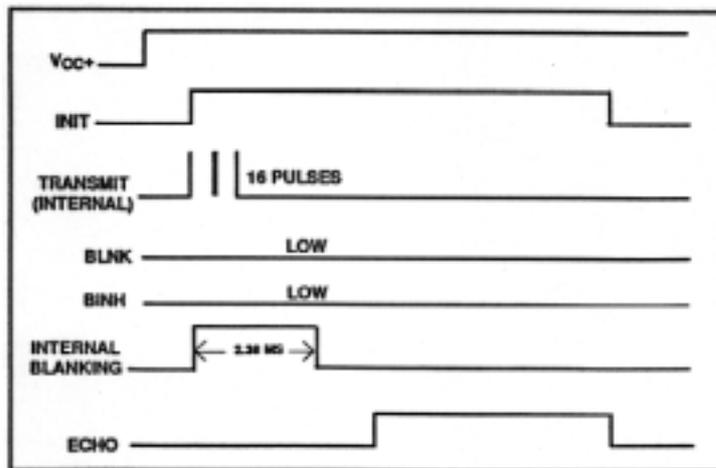


그림 15. 초음파 센서 타이밍 다이어그램

초음파 모듈의 구동원리는 처음에 INIT 신호가 Enable 되면 초음파가 송신되면서 XMIT에 16개의 펄스가 발생한다. 이때 초음파 모듈은

송신모드에서 수신모드로 전환되고 발신된 초음파가 물체에 반사되어 돌아와 센서에 감지되면 장애물과의 거리에 상응되는 ECHO 신호가 초음파 모듈에서 출력된다. 그리고 내부의 증폭기를 통하여, 시간이 지남에 따라 약해진 초음파를 증폭시킴으로써 초음파의 강도를 보상한다. 초음파는 약 344m/s의 음속으로 공기 중에 전파된다. 따라서 초음파가 t초 후에 되돌아온 경우 장애물과의 거리는 식과 같다.

$$L = \frac{344t}{2} = 172 \times t [m] \quad (2.8)$$

이러한 초음파의 원리를 이용하여 송신과 수신사이의 시간을 측정하면 장애물과의 거리를 측정할 수 있다.

2.3.2.2 스테레오 비전(Stereo Vision)

이동로봇은 CCD 카메라를 사용하여 주위환경을 인식하여 작업 대상물 또는 주변에 존재하는 장애물의 2차원 또는 3차원의 위치 또는 거리 정보를 얻는다. 그러나 3차원 공간에 존재하는 물체에 대한 영상을 2차원의 CCD 카메라를 이용하여 포착하면 차원의 감소로 인하여 정보의 손실이 발생한다. 이는 하나의 2차원 화상만으로는 임의의 위치에 놓여 있는 물체까지의 거리정보를 얻을 수 없기 때문이다. 따라서 이동로봇에서 비전센서를 이용하여 거리를 측정하기 위해서는 일반적으로 스테레오 비전을 사용하거나 레이저를 이용하여 이들 사이의 기하학적 관계를 이용한 삼각측량 방식을 많이 사용한다. 본 논문에서 제작한 이동로봇 시스템에서는 스테레오 비전을 이용하여 주위환경을 인식하거나 물체와의 거리를 측정한다.

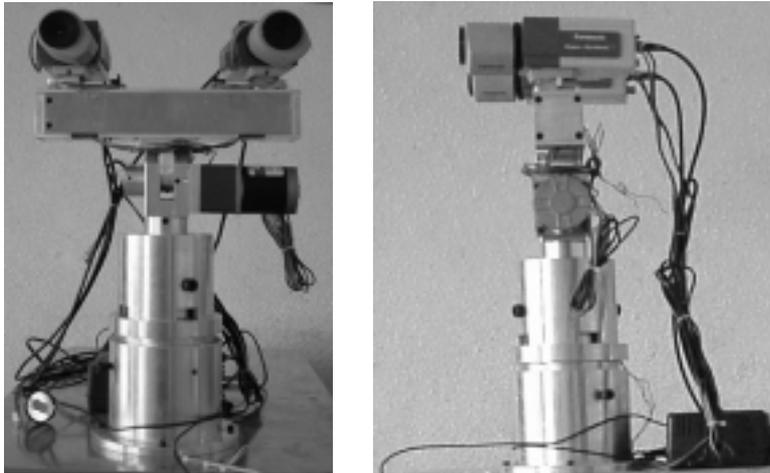


그림 16. 스테레오 비전

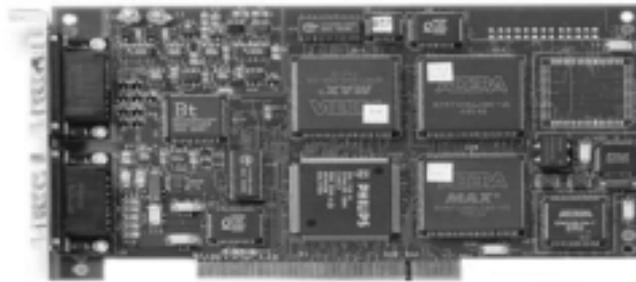


그림 17. 프레임 그레이버

스테레오 비전은 영상을 획득하기 위한 두 대의 CCD 카메라, 획득된 영상으로부터 물체 감지 또는 거리측정을 위해 필요한 정보만을 추출하기 위한 프레임 그레이버, 그리고 스테레오 비전에 자유도를 부여하기 위한 DC 모터 2개와 스텝핑 모터 2개, 포텐셔 미터 2개로 구성되어 있다. 또한 Zoom 렌즈를 통해 원거리의 물체를 인식할 수 있다. 이러한 스테레오 비전은 실제 사람의 눈의 움직임을 흉내낸 것이다. 또한 모터를

구동하기 위해 마이크로 프로세서인 80C196을 사용하여 위치제어를 하였다. 그림 16은 실제 완성된 스테레오 비전을, 그림 17은 프레임 그레이버를 나타낸다.

제3장 인터넷 기반 이동로봇 시스템에서의 시간지연

3.1 인터넷 기반 이동로봇 시스템

일반적인 인터넷 기반 이동로봇 시스템의 전체 구조는 그림 18과 같다. Local Site의 조작자가 Remote Site의 이동로봇을 원격제어하기 위해 인터넷을 통해 명령을 전송하고, Remote Site의 이동로봇은 Local Site의 조작자로부터 받은 명령을 수행하며 이때의 이동로봇의 정보를 다시 Local Site의 조작자에게 인터넷을 통하여 전송한다.

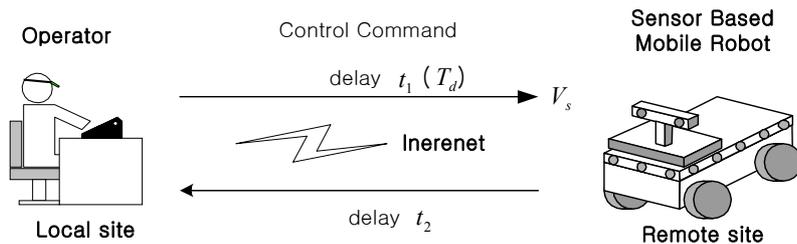


그림 18. 인터넷 기반 이동로봇 시스템의 전체 구조

인터넷을 기반으로 하는 이동로봇의 원격제어 시 다수가 관리하고 공유하는 회선을 이용함으로써 여러 가지의 문제점들이 제기될 수 있다. 그중 가장 중요한 문제점중의 하나는 통신의 신뢰성 문제로써, 불특정 다수의 접속자가 통신망을 공유함으로써 인해 통신의 트래픽을 정확히 예측할 수 없는 인터넷 기반의 통신방식에서 발생하는 통신두절 또는 시간지연은 시스템에 커다란 영향을 끼친다. 이는 이동로봇의 특성상 다양한

환경에서의 여러 가지 작업 및 동작이 가능함으로 인해, 오동작의 발생 시 상황에 따라 치명적인 문제를 야기할 수 있다. 이러한 상황을 극복하기 위해 조작컴퓨터와 이동로봇을 온라인상의 인터넷 타임 서버를 이용한 절대시간으로 동기화 하거나 오프라인 상에서 두 시스템을 시간동기화 한 후 정확한 시간지연을 측정하면 시간지연에 따른 적절한 보상을 하기가 용이하게 된다. 즉 아래의 식 (3.1)과 같이 이동로봇이 명령을 받은 시간(T_m)과 전송된 명령에 포함되어온 명령을 내린 시점의 시간(T_c)을 비교함으로써 명령이 전송될 때까지의 시간지연(T_d)을 알아낼 수 있다.

$$T_d = T_m - T_c \quad (3.1)$$

본 논문에서는 다음과 같은 가정 하에서 인터넷 기반 이동로봇의 경로 및 시간을 보상하여 제어한다.

- ① 로봇에 내려지는 경로 명령은 샘플링 시간(T_s)마다 주어진다.
- ② 보다 원활한 통신을 위하여 원격지 조작 컴퓨터와 이동로봇 간의 데이터 이동량을 최소화하기로 한다. 따라서 이동로봇에 내리는 명령은 로봇의 이동방향(\vec{C})만을 지시하고, 이동로봇의 상태는 피드백 받지 않는다.

3.2 시간지연의 분류

인터넷을 통해 이동로봇을 원격제어 할 경우 시간지연이 발생하지 않

는다면 이동로봇은 조작자의 명령대로 경로를 주행하게 된다. 그러나 인터넷상에서의 시간지연은 대부분 필연적으로 발생하며, 예측이 불가능하기 때문에, 인터넷 기반 원격제어 시 발생하는 예측 불가능한 시간지연은 이동로봇 시스템에 매우 큰 영향을 미친다. 이는 시간지연이 발생하면 이동로봇이 조작자가 명령한 경로를 따라 주행하지 못하고 지시 경로를 벗어나게 되며 조작자의 지시 경로와 이동로봇이 실제로 주행한 경로 사이에는 경로오차가 발생하게 된다. 그림 19는 이동로봇의 원격제어 시 발생하는 문제점인 시간지연에 따른 경로오차와 시간오차를 나타낸다. 이때 발생하는 경로오차와 시간오차를 최소화 시켜야만 이동로봇은 조작자가 지시한 경로에 근접하여 주행할 수 있게 된다.

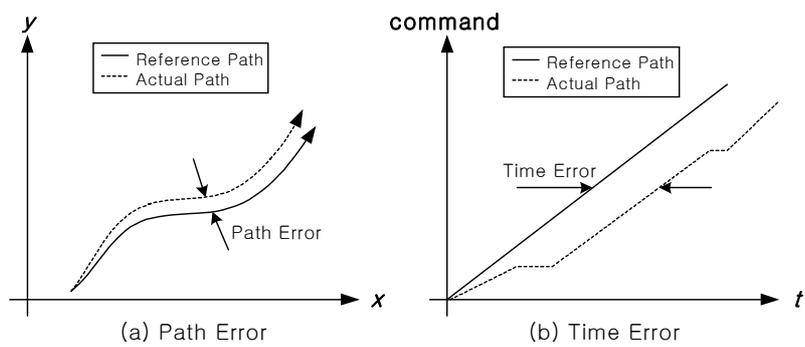


그림 19. 이동로봇의 경로오차와 시간오차

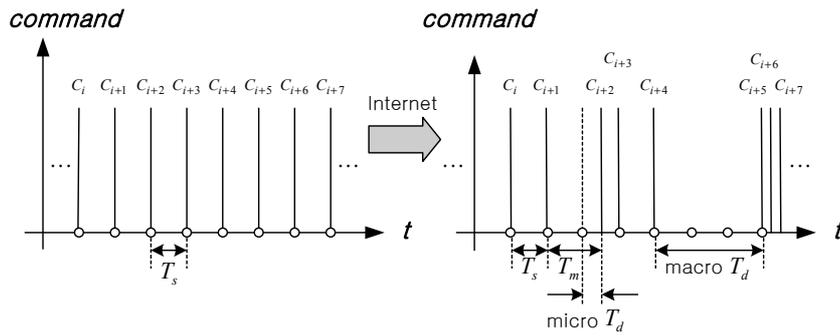


그림 20. 인터넷 기반에서의 시간지연 정의

본 논문에서는 그림20과 같이 시간지연을 크기에 따라 두 가지로 분류하였다. 시간지연이 샘플링 시간(T_s)보다 작다면 Micro Time Delay, 크다면 Macro Time Delay라 정의한다. 즉, Micro Time Delay는 시간지연에 의해 샘플링 시간에 명령을 받지 못했지만 두 번째 샘플링 시간 이전에 지연된 명령을 받게 되는 경우를 의미하고, Macro Time Delay는 두 번째 샘플링 시간이 경과되어도 지연된 명령을 받지 못하는 경우를 의미한다. 이동로봇의 속도가 저속일 경우 Micro Time Delay가 발생한다면 이동로봇이 시간지연 동안 이동하는 거리는 매우 짧다. 따라서 이동로봇은 계획된 경로를 많이 벗어나지 않게 된다. 하지만 이동로봇의 속도가 고속이라면 짧은 시간에 많은 경로를 주행할 수 있으므로 Macro Time Delay 뿐만 아니라 Micro Time Delay가 발생하더라도 계획된 경로를 상당히 많이 벗어나 이동로봇의 경로계획에 큰 영향을 미치게 된다. 이는 컴퓨터 및 주변기기들의 발달과 이동로봇 사용환경의 변화로 이동로봇의 속도가 고속화되어 가는 추세를 고려할 때 Micro Time Delay에 대한 보상은 중요한 문제가 됨을 의미한다.

제4장 인터넷 기반 이동로봇의 시간지연 보상 알고리즘

4.1 인터넷 기반 이동로봇의 시간지연 보상의 개념

그림 21은 본 논문에서 제안하는 인터넷 기반 이동로봇의 경로 및 시간보상 개념의 전체 구조를 나타낸다.

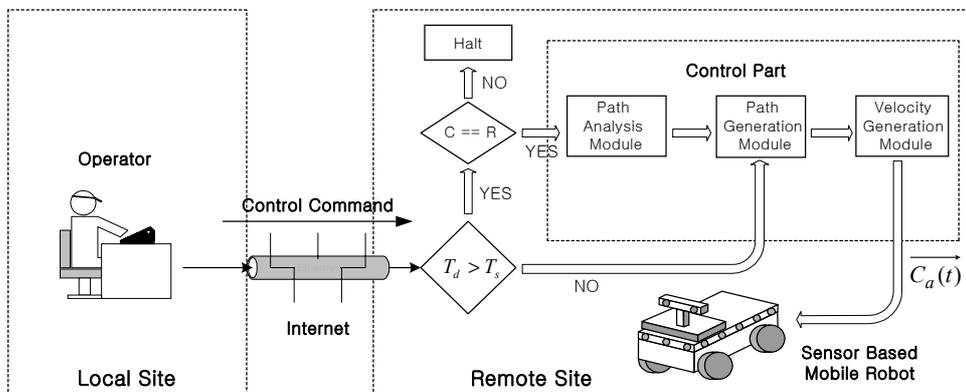


그림 21. 인터넷 기반 이동로봇의 경로 및 시간보상 전체 구조

Local Site의 조작자가 인터넷을 통하여 Remote Site의 이동로봇에게 제어 명령을 보낸다. 이때 발생한 시간지연(T_d)이 샘플링 시간(T_s)보다 크다면, 즉 Macro Time Delay라면, 다음 명령(C)을 수신(R)할 때까지 이동로봇은 정지하고, 다음 명령(C)을 수신(R)하였다면, 즉 $C==R$ 라면, 경로해석기(Path Analysis Module)에서 지연된 시간에 맞게 다시 생성된 경로명령을 경로생성기(Path Generation Module)의 입력으로 사용한다. 만약 시간지연이 샘플링시간보다 작다면, 즉 Micro Time Delay라면,

인터넷을 통해 들어오는 이동로봇의 경로명령은 경로생성기의 입력으로 사용된다. 경로생성기에서는 제안된 경로 보상 알고리즘을 사용하여 이동로봇의 경로오차를 보상한다. 경로생성기에서 생성된 경로는 다시 속도생성기(Velocity Generation Module)의 입력으로 사용된다. 속도생성기에서는 이동로봇의 속도를 생성시켜 두 Site간에서 발생한 시간오차를 보상한다.

4.2 퍼지논리를 이용한 인터넷 기반 이동로봇의 시간지연 보상

본 논문에서는 시간지연 발생시 경로오차 발생에 대한 상황을 그림 22와 같이 설정하였다.

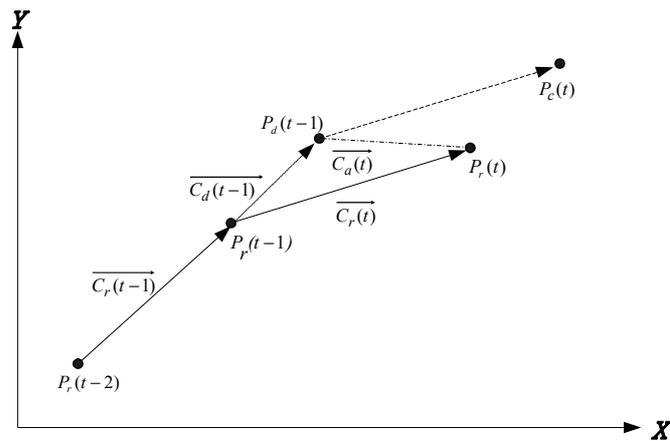


그림 22. 시간지연 발생시 경로오차에 대한 상황

$P_r(t-2)$ 는 이동로봇이 기존의 명령을 받은 지점이고, $P_r(t-1)$ 은 샘플

플링 시간내에 다음 경로 명령을 받아야 하는 지점, 그리고 $P_d(t-1)$ 은 이동로봇이 시간지연에 의해 실제로 명령을 받은 지점이다. $P_r(t)$ 는 $P_r(t-1)$ 에서 시간지연 없이 명령을 받았을 경우 샘플링 시간동안 주행하여 도착해야 하는(다음 명령을 받아야 하는) 지점이고 $P_c(t)$ 는 시간지연 발생 시 적절한 보상이 없는 경우 이동로봇이 샘플링 시간동안 주행 후 도착하게 될 지점이다.

$C_d(t)$ 는 적절한 경로보상이 적용된 경우 로봇이 주행하게 될 최단거리이다. 하지만 이 점에서 이동로봇은 급격한 방향전환을 하게 된다. 이동로봇의 거동을 고려할 때 이러한 급격한 방향전환은 이동로봇에게 치명적인 영향을 미치므로 적절한 경로 보상을 하여 이동로봇의 주행경로를 부드럽게 만들어야 한다.

따라서 본 논문에서는 시간지연이 발생 시 이동로봇이 주행해야 할 경로를 그림 23과 같이 생성시킴으로써 이동로봇의 주행 경로 오차를 줄이고 부드러운 경로를 생성시키는 방법을 제안한다.

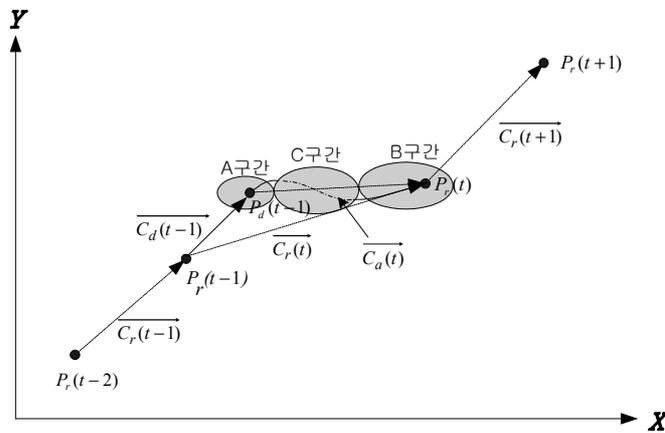


그림 23. 제안된 이동로봇의 주행 경로

이동로봇이 A구간의 점 $P_d(t-1)$ 에서는 $\overrightarrow{C_d(t-1)}$ 의 접선 방향에 가깝게 주행을 하고, B구간의 점 $P_r(t)$ 점에서는 $\overrightarrow{C_r(t)}$ 의 접선 방향에 가깝게 주행을 하여 급격한 방향전환이 발생하지 않도록 한다. C구간은 A구간과 B구간을 부드럽게 연결하는 구간이다.

4.2.1 A구간 경로 생성 알고리즘

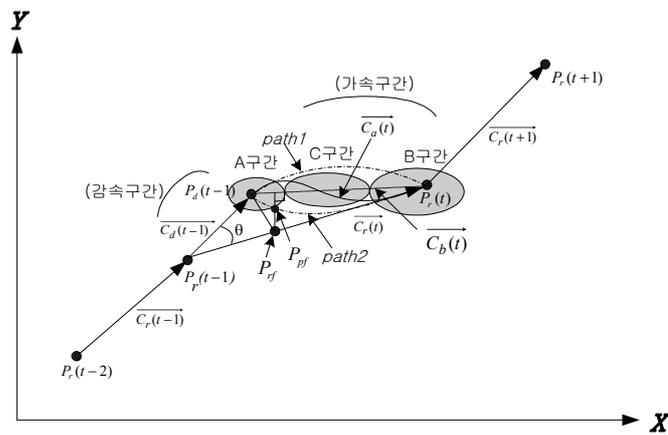


그림 24. 퍼지논리를 이용한 이동로봇의 경로 보상

본 논문에서는 A구간에서 $\overrightarrow{C_d(t-1)}$ 에 접근하는 경로 생성 알고리즘으로 Cubic Spline 보간법을 선택하였다. 그림 24에서와 같이 $P_r(t-1)$, $P_d(t-1)$, $P_r(t)$ 의 세 포인트를 이용하여 Cubic Spline 보간을 함으로써 $\overrightarrow{C_d(t-1)}$ 에 접근하는 경로 Path1을 생성시키고 A구간 생성에는 Path1의 시작부분에 가중치를 주어 사용하였다.

4.2.2 B구간 경로 생성 알고리즘

B구간 경로 생성 알고리즘으로는 $\overrightarrow{C_r(t)}$ 에 접근하는 경로인 Path2를 생성시키고 이때 생성된 Path2의 끝 부분에 가중치를 주어 사용하였다. Path2는 $P_d(t-1)$, $P_r(t)$ 를 포함하는 Cubic Spline으로써, 이를 생성하기 위한 중간점 P_{rf} 를 다음과 같이 선정하였다.

Path2의 중간점 P_{rf} 의 위치선정은 이동로봇의 경로오차에 아주 큰 영향을 미친다. 이동로봇이 계획된 경로를 벗어났을 경우 점 P_{rf} 의 위치에 따라서 경로오차의 크기가 결정되기 때문이다. 하지만 이동로봇의 자세나 주행 경로 등 여러 가지 주변상황에 대해 적절한 점 P_{rf} 의 위치를 선정하기가 쉽지 않다. 따라서 본 논문에서는 이러한 특성에 알맞은 퍼지논리를 이용하여 점 P_{rf} 의 위치를 적절하게 선정하고자 한다. P_{rf} 생성 알고리즘의 입력은 이동로봇이 시간지연동안 이동한 거리와 계획된 경로와 실제경로사이의 각도(θ)이고, 출력은 계획된 경로상의 한 점 P_{rf} 이다.

그림 25는 P_{rf} 생성 알고리즘의 입력변수인 이동로봇이 시간지연동안 이동한 거리에 대한 소속함수이며, 그림 26은 계획된 경로와 실제경로사이의 각도에 따른 소속함수이다. 또한 그림 27은 P_{rf} 생성 알고리즘의 출력변수인 계획된 경로상의 한 점 P_{rf} 를 결정하기 위한 소속함수이다. 이때 전건부의 첫 번째 입력변수와 후건부 출력변수의 입력범위는 이동로봇이 한 샘플링 시간당 최대로 이동할 수 있는 거리이다.

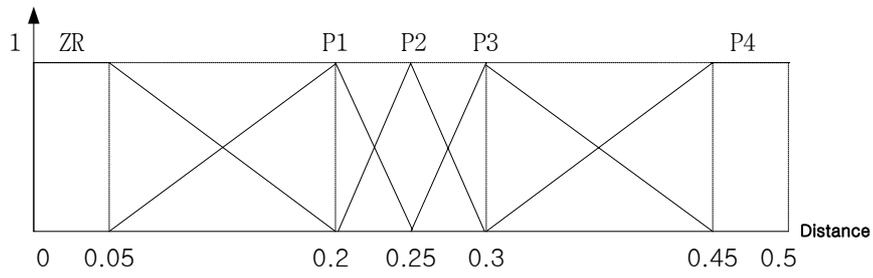


그림 25. 이동로봇의 시간지연 동안 이동한 거리에 대한 소속함수

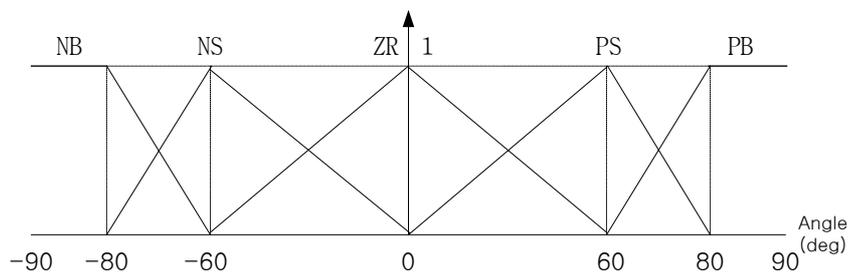


그림 26. 계획된 경로와 실제경로 사이의 각도에 따른 소속함수

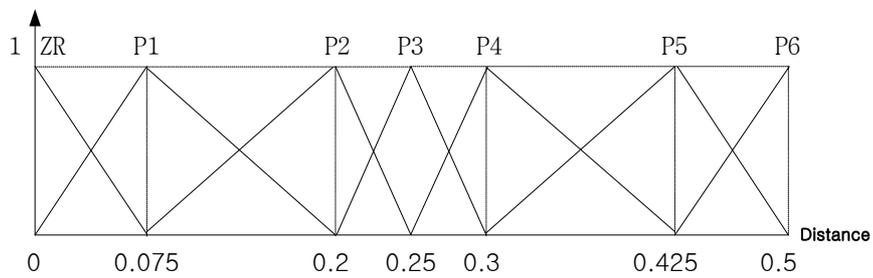


그림 27. P_{rf} 에 대한 소속함수

Dist \ Angle	NB	NS	ZR	PS	PB
ZR	ZR	P1	ZR	P1	ZR
P1	P1	P3	P2	P3	P1
P2	P2	P4	P3	P4	P2
P3	P3	P5	P4	P5	P3
P4	P3	P6	P5	P6	P3

표 3. P_{rf} 생성 알고리즘의 Rule Table

표 3은 P_{rf} 생성 알고리즘에 대한 퍼지규칙이다. 이동로봇이 시간지연동안 이동한 거리가 짧고 두 경로간의 각도가 작으면 이동로봇이 실제 이동해야 할 P_{rf} 의 위치는 $P_r(t-1)$ 에 아주 가깝고, 이동한 거리가 길고 두 경로간의 각도가 크다면 이동로봇이 실제 이동해야 할 P_{rf} 의 위치는 $P_r(t-1)$ 과 $P_r(t)$ 사이의 중앙에 위치되게 함으로써 이동로봇의 경로오차를 줄여 이동로봇이 효율적으로 주행하게 하는 퍼지규칙이다. 여기서 생성된 P_{rf} 와 $P_d(t-1)$, $P_r(t)$ 의 세 포인트를 이용하여 Cubic Spline 보간을 함으로써 Path2를 생성시키면 Cubic Spline 보간의 특성 때문에 특정 구간에서 이동로봇이 계획된 경로를 더 많이 벗어날 수가 있다. 이러한 문제점을 해결하기 위해서 점 P_{rf} 에서 $\overrightarrow{C_b(t)}$ 에 수선을 내리고 이 직선을 이등분하는 점 P_{bf} 를 구하여 Cubic Spline의 입력변수로 사용한다. 즉 $P_d(t-1)$, P_{bf} , $P_r(t)$ 의 세 포인트를 이용하여 Cubic Spline 보간을 함으로써 $\overrightarrow{C_r(t)}$ 에 접근하는 경로인 Path2를 생성시킨다.

4.2.3 C구간 경로 생성 알고리즘

C구간 경로 생성 알고리즘은 Cubic Spline 보간에 의해 생성된 Path1과 Path2의 두 경로를 퍼지조합에 의해 새로운 이동로봇의 경로좌표인 $C_a(t)(x_i, y_i)$ 를 생성시키는 알고리즘으로 이동로봇의 경로오차를 줄임과 동시에 유연한 경로를 생성시킬 수 있다. 여기서 $C_a(t)(x_i)$ 는 $P_d(t-1)$ 에서 $P_r(t)$ 까지 생성되는 x 축 경로좌표이고, $C_a(t)(y_i)$ 는 이에 대응되는 y 축 경로좌표로써 식 (4.1)과 같이 정의된다.

$$C_a(t)(y_i) = Path2(y_i) + (Path1(y_i) - Path2(y_i)) \cdot \alpha \quad (4.1)$$

여기서, α 는 퍼지제어기의 출력값이며, $i = 0, 1, 2, \dots, n$ 으로 $P_d(t-1)$ 에서 $P_r(t)$ 까지 생성되는 경로좌표의 개수이다.

C구간 경로 생성 알고리즘의 퍼지제어기는 후건부가 상수인 TS 퍼지 제어기로써, 입력변수는 이동로봇이 명령을 수신한 점 $P_d(t-1)$ 와 $P_r(t)$ 사이의 x 좌표 값이며, 후건부 출력 값은 0에서 1사이의 값을 가진다.

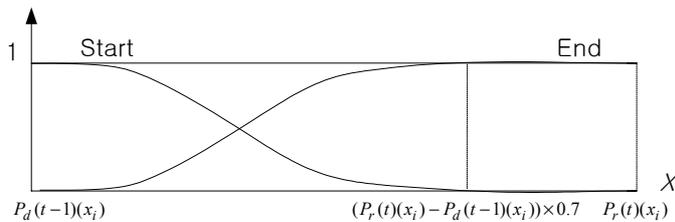


그림 28. $P_d(t-1)$ 에서 $P_r(t)$ 까지 생성되는 x 축 경로좌표에 대한 소속함수

그림 28은 퍼지제어기의 입력변수인 이동로봇이 명령을 수신한 점 $P_d(t-1)$ 와 원격지 조작자가 이동로봇에게 명령한 계획된 경로 $P_r(t)$ 사이에 생성되는 x 축 경로좌표에 대한 소속함수로써 가우시안 함수를 사용하였으며, 퍼지제어기에 사용된 퍼지규칙은 다음과 같다.

If X is Start, Then α is 1

If X is End, Then α is 0

따라서 이동로봇은 $P_d(t-1)$ 에서 다음 경로 명령을 수신한 후, 다음 이동지점인 $P_r(t)$ 를 향해 새로 생성된 경로인 $\overrightarrow{C_a(t)}$ 를 따라 주행하게 되고, 이로써 인터넷 기반 원격제어 시 시간지연에 의해 발생할 수 있는 문제점인 경로오차를 줄이고 동시에 이동로봇의 경로 상에서 발생할 수 있는 급격한 방향전환을 막을 수 있다.

4.2.4 시간오차 보상 알고리즘

Local Site의 조작자가 인터넷을 통하여 Remote Site의 이동로봇을 제어할 경우 시간지연에 의해서 경로오차 뿐만 아니라 시간오차도 발생한다. 시간오차를 줄이기 위해서 경로생성기로부터 생성된 이동로봇의 경로에 근거하여 이동로봇의 속도를 조절한다. 그림 24에서와 같이 이동로봇이 계획된 경로를 따라 주행하다가 $P_r(t-1)$ 지점에 도착할 때까지 시간지연으로 인해 다음 경로명령을 수신하지 못했다면 이동로봇은 속도를 줄여서 최저속도로 지시경로 방향으로 이동하게 된다. 최저속도로 주행하던 이동로봇이 $P_d(t-1)$ 지점에서 다음 경로명령을 수신하였다면

이동로봇은 실제 $P_r(t)$ 지점의 실제 도착시간을 계산한 후 이동로봇의 속도를 가속시켜 시간오차를 줄인다. 그림 29 (a)는 시간오차를 줄이기 위한 속도 프로파일로써, 시간지연이 길게 발생하여 시간지연이 클 경우 이동로봇은 최고속도(V_{max})까지 가속하거나 최저속도(V_{min})까지 감속을 하게 된다. 만약 시간지연이 짧게 발생하여 실제 이동로봇의 속도가 최고속도(V_{max})나 최저속도(V_{min})에 도달하지 않는 경우 V_{max_i} , V_{min_i} 를 이동로봇의 새로운 최고속도와 최저속도로 정의한다.

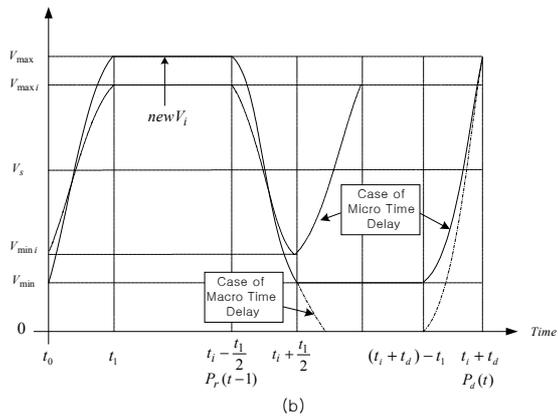
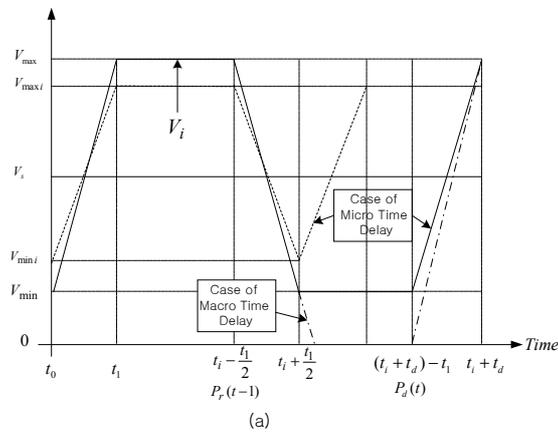


그림 29. 이동로봇의 속도 프로파일

만약 시간지연이 많이 발생해서 이동로봇이 계획된 경로를 많이 벗어났을 경우, 시간오차를 줄이기 위해서는 짧은 시간에 많은 거리를 이동하여야 한다. 이때 이동로봇은 급가감속을 하게 되는데, 이동로봇의 거동을 고려할 때 급격한 가감속은 이동로봇에게 부담이 될 수 있다. 따라서 그림 29(b)는 급격한 가감속을 줄이기 위한 속도 프로파일로써, 이와 같이 부드러운 속도 명령을 주어 이동로봇의 급가감속을 방지한다. 이를 위한 알고리즘으로 식 (4.2)와 같이 표현된다.

$$newV_i = V_{\min i} + (V_{\max i} - V_{\min i}) \cdot \beta \quad (4.2)$$

여기서, $newV_i$ 는 퍼지로직에 의해 생성된 속도이고, $V_{\max i}, V_{\min i}$ 는 이동로봇의 실제 최고속도와 최저속도, β 는 퍼지제어기의 출력값, $i=0,1,2,\dots,n$ 이다.

퍼지로직을 이용한 속도 프로파일 생성 알고리즘의 퍼지제어기는 후건부가 상수인 TS 퍼지제어기로써, 입력변수는 이동로봇의 가감속 시간 t_1 이고, 후건부 출력값은 0에서 1사이의 값을 가진다.

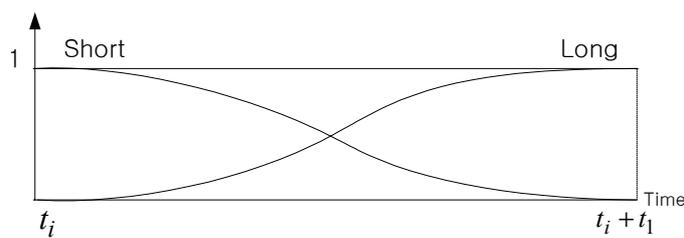


그림 30. 이동로봇의 가감속 시간에 대한 소속함수

그림 30은 속도 프로파일 생성을 위한 퍼지제어기의 입력변수인 이동 로봇의 가감속 시간에 대한 소속함수로써 가우시안 함수를 사용하였다. 속도 프로파일 생성에 사용된 퍼지규칙은 다음과 같다.

If Time is Short, Then β is 0

If Time is Long, Then β is 1

위에서 설명한 바와 같이 이동로봇의 인터넷 기반 원격제어 시 시간 지연이 발생한 경우 시간지연이 이동로봇에게 미치는 영향을 줄이기 위해 퍼지논리를 이용한 인터넷 기반 이동로봇의 시간지연 보상 알고리즘을 적용시킴으로써 시간지연에 대해 적절한 보상을 할 수 있다.

제5장 시뮬레이션

5.1 시뮬레이션을 위한 파라미터

본 시뮬레이션에서는 이동로봇이 주행 중 시간지연에 의해 다음 경로 명령을 샘플링 시간 내에 수신하지 못했을 경우, 본 논문에서 제안한 퍼지논리를 이용한 인터넷 기반 이동로봇의 시간지연 보상 알고리즘에 대해 시뮬레이션을 통하여 성능을 확인해 보고자 한다. 우선 샘플링 시간 T_s 는 0.5초로 설정하였고 시간지연 T_d 에 있어서는 Micro Time Delay는 0.5초 미만으로 Macro Time Delay는 0.5초 이상으로 설정하였다. 따라서 본 시뮬레이션에는 Micro Time Delay와 Macro Time Delay 두 가지가 복합적으로 발생하는 시간지연에 대해 시뮬레이션 하였다. 그리고 이동로봇의 속도 V_s 는 1 m/s로 설정하였다. 아울러 표 4는 시뮬레이션에 사용된 경로 명령 시간(T_c)과 시간지연을 가진 명령을 이동로봇이 수신한 시간(T_m)을 나타내고 있다. 그림 31에는 표 4에서 제시된 시뮬레이션에 사용된 시간지연을 도식화하였다.

명령	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
T_c	0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5	6.0	6.5	7.0
T_m	0.0	0.67	1.30	1.75	2.20	2.83	3.26	3.90	5.37	5.37	5.37	5.77	6.37	6.89	7.0
T_d	0.0	0.17	0.30	0.25	0.20	0.33	0.26	0.40	1.37	0.87	0.37	0.27	0.37	0.39	0.0

표 4. 시뮬레이션에 사용된 시간테이블

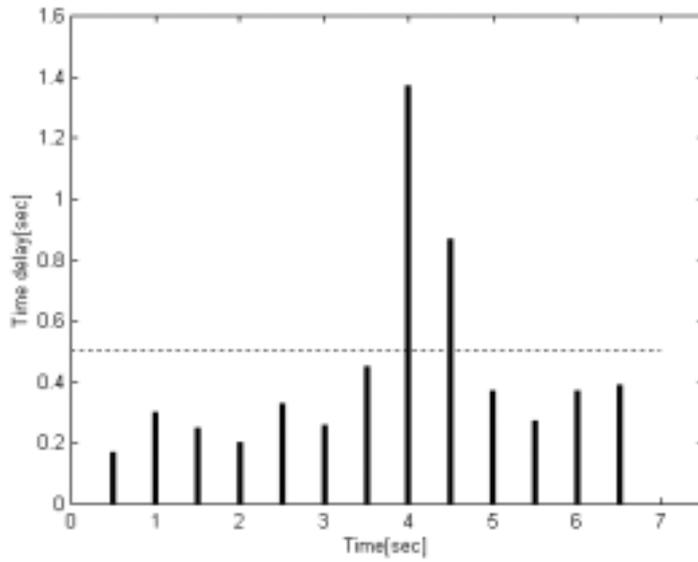


그림 31. 시뮬레이션에 사용된 시간지연

5.2 시간지연 발생 시 적절한 보상이 없는 경우

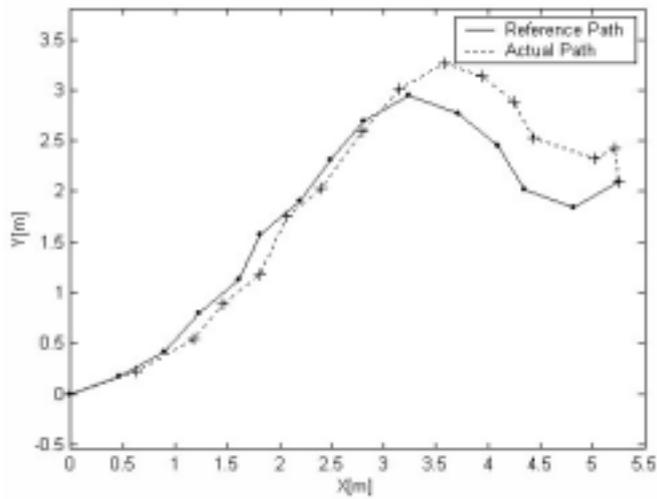


그림 32. 시간지연 발생 시 적절한 보상이 없는 경우

그림 32는 시간지연 발생 시 적절한 보상이 없는 경우에 시간지연이 이동로봇의 경로에 미치는 영향을 나타낸다. 경로 보상을 하지 않는 경우 이동로봇은 시간지연에 따라서 계획된 경로를 상당히 많이 벗어나게 된다.

5.3 시간지연에 대한 적절한 보상이 적용된 경우

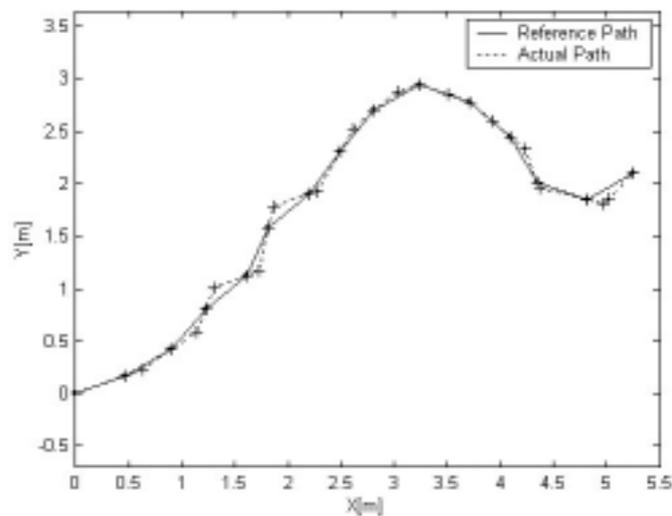


그림 33. 시간지연에 따른 적절한 보상이 적용된 경우

그림 33은 인터넷 기반 원격제어 시 발생하는 시간지연에 따른 적절한 보상이 적용된 경우이다. 이 경우에는 이동로봇은 계획된 경로를 잘 추종하고 있지만 급격한 방향전환을 하는 것을 볼 수 있다.

5.4 퍼지논리를 이용하여 시간지연을 보상한 경우

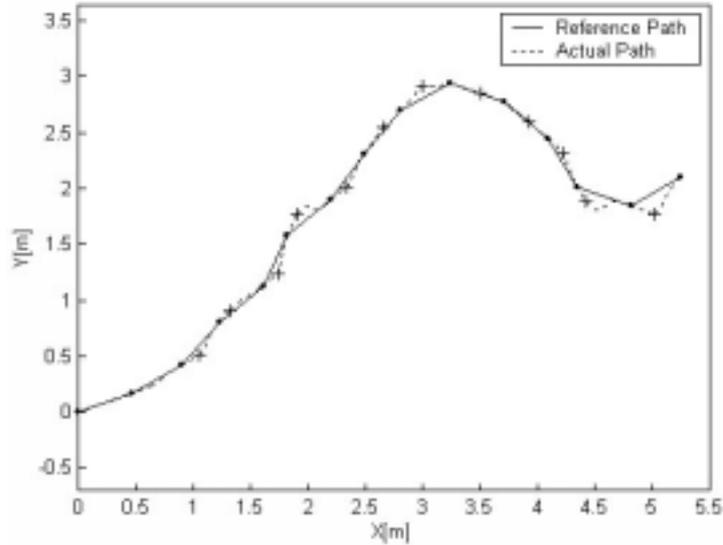


그림 34. 퍼지논리를 이용하여 시간지연 보상을 한 경우

그림 34는 퍼지논리를 이용하여 시간지연 보상을 한 경우이다. 경로 오차가 줄었을 뿐만 아니라 이동로봇 경로상의 급격한 방향전환도 제거 되었으므로 시간지연이 발생하더라도 이동로봇이 계획된 경로를 잘 추종함을 볼 수 있다.

아울러 그림 35는 시뮬레이션에 사용된 이동로봇의 속도 프로파일이 다. 이동로봇의 속도를 변화시켜 시간오차를 줄였고 이동로봇이 주행 중 급가감속을 방지하기 위하여 퍼지논리를 이용하여 속도 프로파일을 생성시켰다.

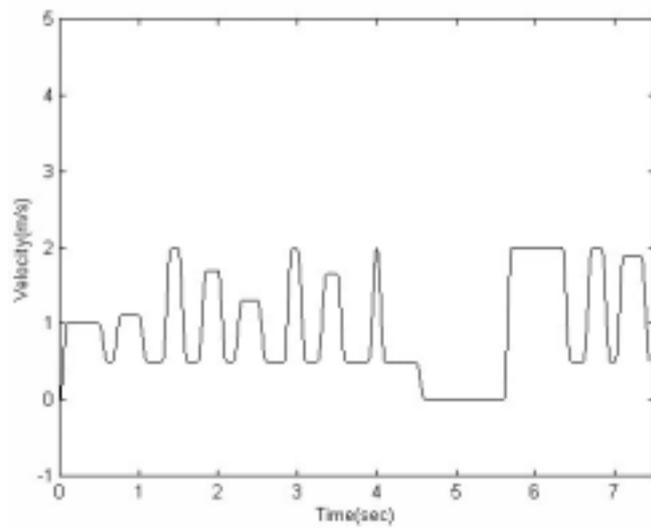


그림 35. 시뮬레이션에 사용된 속도 프로파일

	시간지연에 따른 보상이 없는 경우	시간지연에 따른 적절한 보상이 적용된 경우	퍼지논리를 이용하여 보상을 한 경우
경로오차	1.1056 m^2	0.3465 m^2	0.3218 m^2
시간오차	1.7015 sec	1.6900 sec	0.4400 sec

표 5. 시간지연 보상에 따른 성능지수 비교

시뮬레이션의 결과를 평가하기 위하여 경로보상에 대한 성능평가 함수로서 계획된 경로와 이동로봇이 주행한 실제경로간의 차이에 의해 생기는 면적을 계산하였다. 또한 시간보상에 대한 성능평가 함수로서는 이동로봇이 목적지까지 주행 시 걸린 시간을 계산하였다. 그 결과는 표 5와 같으며 이동 로봇의 주행 경로 및 주행 시간이 개선되었음을 볼 수 있다.

제6장 결론

본 논문에서는 인터넷을 기반으로 한 이동로봇의 원격제어 시 발생하는 시간지연이 이동로봇의 주행에 미치는 영향을 최소화하기 위해 퍼지논리를 이용한 인터넷 기반 이동로봇의 시간지연 보상을 제안하였고, 시뮬레이션을 통하여 제안된 알고리즘을 타당성을 검토해 보았다. 시간지연에 대한 보상이 없는 경우 이동로봇은 계획된 경로를 상당히 많이 벗어나게 되고, 시간지연에 따른 적절한 보상을 할 경우 경로오차는 줄일 수 있지만 시간지연이 발생한 후 다음 명령을 실행할 때 이동로봇이 급격한 방향 전환을 하게 된다. 본 논문에서 제안한 퍼지논리를 이용한 이동로봇의 시간지연 보상에서는 이동로봇의 급격한 방향전환을 방지하여 부드러운 주행경로를 생성함은 물론 경로오차를 최소화함으로써 이동로봇이 더욱 더 안정적으로 주행함을 알 수 있다. 또한 이동로봇의 속도를 변화시켜 두 사이트간의 시간오차를 줄였고 퍼지논리를 이용하여 이동로봇이 급가감속을 보상하였다.

참 고 문 헌

- [1] K. Goldberg, M. Maschna, S. Gentner, et al., *Desktop teleoperation via the www*, Proceedings of the IEEE International Conference on Robotics and Automation, pp. 654-659, Japan 1995.
- [2] <http://telegarden.aec.at>
- [3] K. Taylor and B. Dalton, *Internet Robotics : A New Robotics Niche*, IEEE Robotics and Automation Magazine, Vol. 7, No. 1, pp. pp. 27-34, March 2000
- [4] D. Schulz, W. Burgard, Dieter Fox, S. Thrun, and A. Cremers, *Web interfaces for Mobile Robots in public places*, IEEE Robotics and Automation Magazine, Vol. 7, No. 1, pp. 48-56, March 2000
- [5] Riko Safaric, Karel Jezernik, D. W. Calkin, R. M. Parkin, *Telerobot Control via Internet*, Proceedings of the IEEE International Symposium on Industrial Electronics, Vol. 1, pp. 298-303, July 1999,
- [6] Imad Elhajj and Ning Xi, Yun-hui Liu, *Real-Time Control of Internet Based Teleoperation with Force Reflection*, Proceedings of the 2000 IEEE International Conference on Robotics and Automation San Francisco. CA, pp. 3284-3289, April 2000
- [7] K. Ohba, S. Kawabata, N. Y. Chong, K. Komoriya, T. Matsumaru, N. Matsuhira, K. Takase, and Tanie, *Remote Collaboration through Time Delay in Multiple Teleoperation*, Proceedings of the 1999 IEEE/RSJ International Conference on Intelligent Robotics and

- Systems, Vol. 3, pp. 1866-1871, October 1999
- [8] Ren C. Luo, Tse Min Chen, *Development of a Multibehavior-Based Mobile Robot for Remote Supervisory Control Through the Internet*, IEEE ASME Transactions on Mechatronics, Vol. 5, No. 4, pp. 376-385, December 2000
- [9] Ren C. Luo, Tse Min Chen, and Chi Yang Hu, *Networked Intelligent Autonomous Mobile Robot: Issues and Opportunities*, IEEE International Symposium on Industrial Electronics, ISIE 99, pp. PS7-PS13
- [10] Nak Young Chong, Tetsuo Kotoku, Kohtaro Ohba, Hisayuki Sasaki, Kiyoshi Komoriya, and Kazuo Tanie, *Use of Coordinated On-line Graphics Simulator in Collaborative Multi-robot Teleoperation with Time Delay*, Proceedings of the 2000 IEEE International Workshop on Robot and Human Interactive Communication Osaka, Japan, pp. 167-172, September 27-29 2000
- [11] Gokiso-cho, Showa-ku, *Network-Based Force-Reflecting Teleoperation*, Proceedings of the 2000 IEEE International Conference on Robotics & Automation San Francisco, pp. 3126-3131, April 2000
- [12] Kensuke Tanaka, Eiichi Nakagawa, Masanori Ito, Naoki Mizuno, Takayoshi Yamada, Etsuro Shimizu, Kazuyuki Kagayama, *An Internet-based Tele-robot Environment for A Time Critical Task*, Proceedings of the 1999 IEEE International Conference on Systems, Man and Cybernetics - Volume 5 ,V.5 ,1999 -10 -12
- [13] A. S. Sekmen, Z. Bingul, V. Hombal, and S. Zein-Sabatto,

- Human-Robot Interaction over the Internet*, Proceedings of the 2000 IEEE Southeastion, 2000 -04 -07
- [14] H Vermeulen, T I Niekerk, J Huang, D Hattingh, *VRML to Monitor and Control an Industrial Robot via the Internet*, Proceedings of the 1999 IEEE Africon 5th Africon Conference in Africa - Volume 1 ,V.1 ,1999 -09 -20
- [15] Roberto Oboe, Stefano Piovan, *Sensorless Force Reflecting Teleoperation for Low Cost Web-interfaced System*, Proceedings of the IEEE International Symposium on Industrial Electronics - Volume 1 ,V.1 ,1999 -07 -12
- [16] 박민용, 김승우, *산업적 활용을 위한 이동로봇 시스템의 개발*, 1996
- [17] 강성철, 정낙영, *인터넷 기반 원격 매니플레이션 : 일본의 연구개발 사례를 중심으로*, ICASE Magazine Vol. 6, No. 6, 2000

Abstract

**Path Compensation Control for Remote
Control of Internet Based Mobile Robot with
Time Delay**

by Bong Soo Yoo

Department of Electrical, Electronic and Control Engineering
Graduate School, Changwon National University
Chagwon, Korea

(Supervised by Professor Joong-Seon Joh)

Recently, the internet technologies have become widely popular and used in various research areas. Especially, as the researches about tele-robot system using the internet technology are actively progressed, the time delay, which caused by state and bandwidth of internet, was become important issues of researches. In order to control the mobile robot, the time delay has to be compensated.

In this paper, the operator sends velocity commands with time data in order to the mobile robot could move according to remote operator's commands. Then the mobile robot calculates the time delay, and compensates the path error and time error according to the time delay. Therefore the mobile robot tracks the appropriated path with minimum time error.

감사의 글

부족함이 많은 저에게 배움의 길을 열어주시고, 새로운 지식들을 접하고 연구하는 방법들에 대해 많은 가르침을 주신 조중선 교수님께 진심으로 감사드립니다. 또한 본 논문을 세밀히 심사해주신 조철우 교수님, 김재형 교수님께 감사드립니다.

학문의 동료이자 제2의 가족인 지능 메카트로닉스 연구실의 여러 선후배님들께 고마움을 전합니다. 부족한 후배를 자상히 이끌어 주시고 선배로서의 귀감을 보여주신 영진 선배, 상민 선배, 정목 선배, 철균 선배, 만석 선배의 고마움을 잊을 수 없습니다. 그리고 같은 연구실에서 생활하면서 여러모로 많은 도움을 준 경엽, 정욱, 기범, 세욱, 한호, 영일, 정현, 은정에게 또한 감사의 마음을 전합니다. 아울러 함께 하면서 항상 격려해 주신 제어계측공학과 선후배님들, 근우 선배, 성학 선배, 태성 선배, 병철 선배, 재석 선배, 달규 형, 형장 형, 재봉 형, 영삼 선배, 대현 선배, 옥우 선배, 교훈 선배, 원철, 성민, 동희, 정섭, 혁민, 순주, 일수, 석진, 용택, 현수, 인화, 광인, 일서, 준호 님에게도 감사드립니다.

힘이 들 때마다 따뜻한 말로 격려를 아끼지 않는 창원전문대학 새마을 동아리 선후배님들, 종관 선배, 상열 선배, 충귀 선배, 세광 선배, 옥만 선배, 세원 선배, 태호, 성국, 현철, 종민, 창호, 덕숙, 혜영, 은영, 혜숙, 경주, 주현, 민식, 민석, 철집, 경재, 현수, 정석, 계현, 수인, 현정, 미옥, 진화, 윤경, 미정, 성욱, 나미, 미화 님들을 비롯한 여러 새인들께 고맙다는 말을 전합니다.

끝으로 언제나 아낌없이 사랑을 베풀어 주시고 말없이 도와 주시는 형님, 형수님, 누나, 매형께도 감사드리며, 이 논문을 사랑하는 어머니께 바칩니다.