

# 이동로봇의 무선 원격조정을 위한 적응제어

심 교 만, 홍 현 주, 권 석 근, 노 영 식  
울산대학교 전기전자정보시스템공학부  
전화 : 052-259-1642 / 핸드폰 : 016-817-3370

## The Adaptive Control for the Teleoperated Mobile Robot through the Wireless LAN

Kyo-Man Shim, Hyun-Ju Hong, Seok-Geun Kwon, Young-Shick Ro  
School of Electrical Engineering, Ulsan University  
E-mail : shim3370@hitel.net

### Abstract

When a system is teleoperated in the indoor environment through the wireless LAN, we must consider the communication time delay. The time delay is random and unbounded due to the inherent characteristic and surrounding environment. As result of this, the stability and performance of the teleoperated system is degraded.

In this paper, we present the adaptive control algorithm to overcome this problem by measuring time delay and allocating task corresponding to time delay dynamically. And the simple experiment is conducted to demonstrate the feasibility.

### I. 서론

이동로봇의 원격제어 시스템은 서버와 클라이언트 그리고 통신채널로 구성된다. 통신은 매체의 형태에 따라 유선과 무선으로 구분할 수 있다. 무선랜(Wireless LAN)을 통한 이동로봇의 무선 원격조정은 유선보다 유연성을 보장할 수 있다. 장애물이 없는 환경에서 무선랜의 통신 두절과 통신지연은 거리에 의해 좌우된다. 그러나 건물 내에서는 거리뿐 아니라 벽과 같은 고정 장애물, 사람과 같은 이동 장애물에 의해 통신이 두절될 가능성과 지연이 불규칙하게 발생할 가능성이 높다. 또한 실시간 모니터링을 위한 영상정보와 센서 데이터의 전송은 통신채널의 사용대역폭을 많이 요구

하게 된다. 이와 같이 제한된 대역폭, 통신지연과 두절은 원격제어시스템을 불안정하게 하고 성능을 크게 저하시킨다.

본 논문은 이러한 문제를 극복하기 위하여 실시간으로 통신 지연과 연결상태를 측정하고 능동적으로 제어 모드를 전환하는 적응제어 알고리즘을 제안하고 활용 가능성을 입증하고자한다.

### II. 원격제어시스템의 구조

원격제어시스템의 구조는 그림 1과 같이 크게 서버와 클라이언트 그리고 통신 채널로 나눌 수 있다. 서버는 이동로봇에 장착된 컴퓨터로 정의하고 클라이언트는 조작자와 인터페이스되는 컴퓨터로 정의한다. 통신 채널은 무선랜으로 구현되었다.

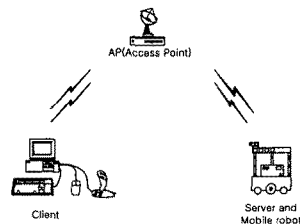


그림 1. 원격제어시스템의 구조

조작자와 이동로봇을 포함한 원격제어시스템의 제어

루프는 그림 2와 같다.

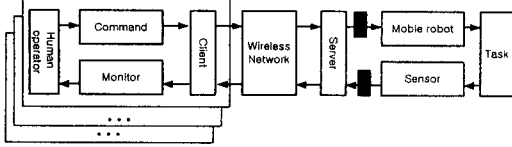


그림 2. 원격제어시스템의 제어루프

### 2.1 서버 시스템

서버 시스템은 그림 3과 같이 PC(Pentium IV 1.4G), 이동로봇(TRC Labmate), 초음파 센서, PC 카메라, 무선 랜으로 구성되어 있으며, 기능적으로 클라이언트의 제어권을 관리하고, 제어명령수행 그리고 작업환경에 대한 영상정보를 접속된 모든 클라이언트에게 전송한다.

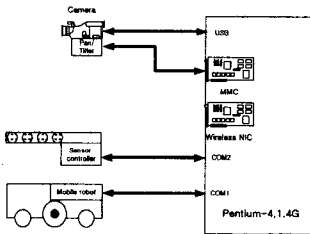


그림 3. 서버 시스템의 구성

### 2.2 서버 소프트웨어 객체 구조

시스템의 효율적 관리를 위해 소프트웨어는 객체화하였다[1]. 전체구조는 센서관리, 통신관리, 요청관리와 제어루프관리 네 개의 객체로 구성되어 있다.

센서관리객체는 카메라 영상정보의 취득과 전처리를 수행하고, 초음파 센서와 근접 센서로부터 장애물 거리정보를 처리한다.

통신관리객체는 표 1과 같이 클라이언트, 이동로봇, 센서 모듈과의 통신을 관리하고 무선망의 통신 지연을 실시간으로 측정한다.

표 5. 서버 시스템의 통신 프로토콜

| 대 상           | 프로토콜   |
|---------------|--------|
| 클라이언트(명령, 상태) | TCP/IP |
| 클라이언트(영상정보)   | UDP/IP |
| 이동 베이스        | COM1   |
| 초음파 센서        | COM2   |
| 근접 센서         | LPT1   |
| PC 카메라        | USB    |

요청관리객체는 클라이언트의 연결과 해제를 담당한다. 동시에 여러 클라이언트가 제어권을 요구하면 인증을 받은 클라이언트인지 확인 절차를 수행하고, 제어권을 부여한다.

제어루프관리는 측정된 통신 지연에 따라 능동적으로 제어모드 전환을 수행한다. 제어루프관리는 본 논문에서 제안한 알고리즘을 실제 구현한 객체로써 제 3장에서 상세히 설명 할 것이다.

### 2.3 클라이언트 소프트웨어 객체 구조

클라이언트는 범용 PC에 조이스틱과 클라이언트용 소프트웨어만 설치하면 서버 시스템에 접속하여 이동로봇을 원격조정 할 수 있다. 클라이언트 객체 구조는 사용자 인터페이스, 통신관리, 요청 관리로 구성되어 있다.

원격조정에 있어서 장치들에 대한 인터페이스는 이용자가 쉽고 편리하게 사용할 수 있고 유연성 있게 설계되어야 한다[2]. 유연성을 고려하여 키보드 인터페이스는 하위 수준의 명령어를 직접 편집할 수 있도록 설계하였고, 화면구성은 원격지환경과 통신 상태를 한 눈에 모니터링 할 수 있도록 하였다. 조이스틱으로 서버 시스템의 카메라를 조정할 수 있으며, 또한 이동로봇의 주행방향과 이동속도를 제어할 수 있다.

## III. 적응제어 알고리즘

### 3.1 실시간 통신망 지연시간 측정

RTT(Round Trip Time;지연시간)는 불규칙하고 무계(Unbounded)이기 때문에 많은 연구들에서처럼 통신지연이 일정하다고 가정할 수 없으며, 또한 확률·통계적으로 모델링 할 수 없다. 따라서 RTT는 실시간으로 측정되어야 하며, 측정된 지연시간에 따라 원격제어시스템을 제어할 필요가 있다.

RTT는 클라이언트측에서 제어명령을 전송하고 궤환되었을 때 측정하는 방법과 서버측에서 시험 데이터(Probe data)를 사용하여 측정하는 방법이 있다. 클라이언트측에서 측정하는 방법은 통신 두절과 데이터가 손실되었을 때 지연시간은 크게 증가 할 것이다. 결과적으로 작업환경과 원격시스템의 상태를 실시간 모니터링 할 수 없게 된다. 따라서 지연이 많은 영상정보를 이용해서 제어를 하는 것은 시스템의 안정도를 보장할 수 없다. 서버측에서 측정하는 방법은 시험데이터를 전송하고 궤환되는 시점을 기다리고 있는 것이 아니라 주기적으로 궤환 유무를 판단함으로써 실시간으로 통

신망의 상태를 측정할 수 있다. 측정된 지연시간에 따라서 능동적으로 제어모드를 전환할 수 있다.

본 논문에서는 그림 4와 식 (1)과 같이 통신망의 RTT를 모델링 하였다.  $\delta(t)$ 는 RTT에 비해 상대적으로 작기 때문에 무시하였다.

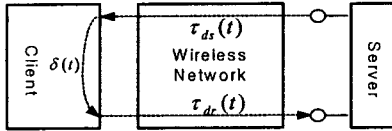


그림 4. 통신망 모델

$$\tau(t) = \tau_{ds}(t) + \tau_{dr}(t) + \delta \cong \tau_{ds}(t) + \tau_{dr}(t) \quad \text{--- (1)}$$

여기서,

$\tau(t)$ : RTT(Round Trip Time)

$\delta(t)$ : 클라이언트에서 시험데이터 처리 시간

$\tau_{ds}(t)$ : 서버에서 클라이언트로 통신 지연 시간

$\tau_{dr}(t)$ : 클라이언트에서 서버로 통신 지연 시간

### 3.2 능동적 제어모드 전환

측정된 RTT에 의한 능동적 제어모드전환 알고리즘은 그림 5와 같다. RTT가 경계값 보다 작으면 제어루프 관리객체에 의해서 직접제어모드로 전환한다. 이때 조작자는 이동로봇의 속도를 제어할 수 있다.

RTT가 경계값 보다 크면 제어루프 관리객체는 행동 제어모드로 전환한다. 경계값은 이동로봇의 주행속도와 영상전송률을 고려하여 결정되어야 한다. 이때 조작자는 이동로봇의 위치제어만 할 수 있고, 속도는 서버에 의해서 제어된다[3].

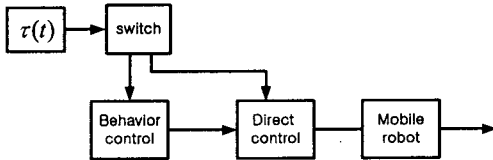


그림 5. 제어모드 전환 블럭도

### 3.3 적응제어 알고리즘

직접제어모드에서 조작자는 조이스틱으로 이동로봇의 속도를 제어한다. 조이스틱의 좌표계와 이동로봇의 속도 좌표계의 관계는 그림 6과 같고 변환관계식은 식 2와 같다. 통신 지연을 고려한 적응제어기는 그림 7과

같고 이동로봇의 속도제어 관계식은 식 (3)과 식(4)와 같다. 적응제어기는 지연시간에 따라 직선 주행속도와 각속도를 비례 감소시키고 최대 지연허용시간( $\tau_{max}$ )을 초과하면 정지하도록 설계하였다.

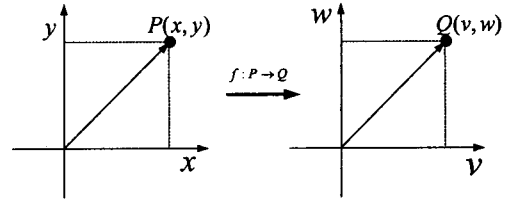


그림 6. 조이스틱 좌표계 변환

$$\begin{aligned} x_{max} &= 400 & y_{max} &= 400 \\ v &: \text{이동 로봇의 직선주행속도(mm/sec)} \\ w &: \text{이동 로봇의 회전각속도(deg/sec)} \end{aligned}$$

$$\begin{vmatrix} v \\ w \end{vmatrix} = \begin{vmatrix} 1 & 0 \\ 0 & 0.05 \end{vmatrix} \begin{vmatrix} x \\ y \end{vmatrix} \quad \text{--- (2)}$$

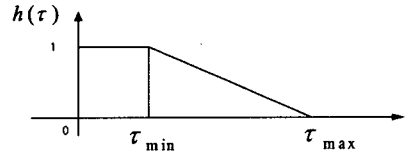


그림 7. 시간 지연을 고려한 적응 제어기

$$\begin{aligned} v' &: \text{시간 지연을 고려한 직선주행속도(mm/sec)} \\ w' &: \text{시간 지연을 고려한 회전각속도(deg/sec)} \end{aligned}$$

$$v' = v \cdot h(\tau(t)) \quad \text{--- (3)}$$

$$w' = w \cdot h(\tau(t)) \quad \text{--- (4)}$$

여기서,

$$\tau(t) \leq \tau_{min} : h(\tau(t)) = 1$$

$$\tau_{min} < \tau(t) < \tau_{max} : h(\tau(t)) = \frac{\{\tau_{max} - \tau(t)\}}{\{\tau_{max} - \tau_{min}\}}$$

$$\tau(t) \geq \tau_{max} : h(\tau(t)) = 0$$

## IV. 원격주행실험

### 4.1 실험 환경

제안된 알고리즘의 안정성과 강건함을 입증하기 위해 정적인 환경이 아닌 동적 환경에서 실험을 실시하였다. 원격조정에 의한 복도 주행과 사무실 출입 등의 실험을 본교 산학협동과 2층 로비와 사무실에서 실시하였다. 이와 같은 환경은 사람과 같은 동적인 장애물

의 존재가능성이 높고, 사무실간의 벽과 같은 장애물에 의해 통신지연과 데이터 손실의 확률이 높으므로 제안된 알고리즘의 유용성을 입증하기에 적합하다고 사료된다. 본 실험에서는 표 2와 같이 영상 데이터의 크기는 160X120에 초당 25frame으로 전송하였고, 이동로봇의 주행속도는 안전성을 고려하여 400mm/sec로 제한하였다.

표 6. 구성요소들의 제어주기

| 영상 전송률[160X120]<br>[frames/sec] | 주행속도<br>[mm/sec] | 초음파 스캔주기<br>[scan/sec] |
|---------------------------------|------------------|------------------------|
| ≤25                             | ≤400             | 2                      |

#### 4.2 실험 결과 고찰

그림 8은 사무실문 통과시 상태를 나타낸다. 그림과 같이 초음파에 의해 장애물의 정보를 획득하고 카메라를 조정하여 다시 한번 확인한 후에 성공적으로 통과를 수행하였다.

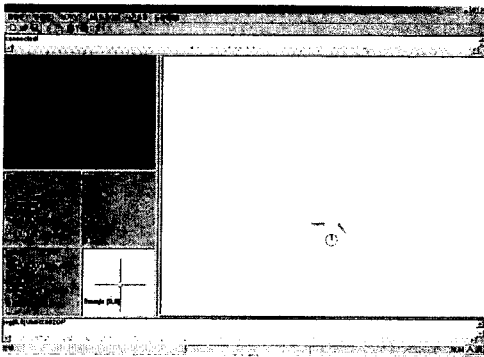


그림 8. 문 통과 실험

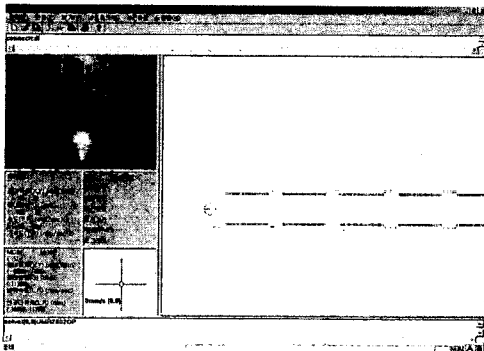


그림 9. 복도 주행 실험

그림 9는 복도주행 실험을 수행한 결과를 나타내고 있다. 그림에서처럼 주위의 장애물 정보도 나타내고 있다. 복도주행 실험에서 서버와 클라이언트간의 거리가 멀어짐에 따라서 영상의 전송률이 크게 나빠졌다. 즉, 지연시간이 증가하게 되었고 이동로봇의 주행속도는 서버에 의해서 능동적으로 감소하였다.

### V. 결론

본 논문은 무선랜을 사용하여 원격제어시스템을 구축하고 서버와 클라이언트를 객체화하여 효율적으로 관리하도록 구현하였다. 사용자 인터페이스는 쉽고 편리하게 사용할 수 있도록 하였다. 적응제어 알고리즘을 구현하기 위하여 서버측에서 실시간으로 RTT를 측정하고, 측정된 RTT에 따라 제어기는 직접제어와 행동제어모드로의 전환이 능동적으로 이루어지도록 설계하였다. 구현된 원격제어시스템에 제안한 알고리즘을 적용하여 실험을 수행하였다.

본 논문은 통신망이 내포하고 있는 통신 지연과 데이터 손실을 극복하고 원격제어시스템을 안정하고 강건하게 하는 알고리즘을 제시하고 그 가능성을 확인하였으며 여러 응용에 활용될 수 있다고 판단된다.

향후 공용망인 인터넷에서 원격제어가 이루어질 수 있도록 이동로봇을 포함한 서버시스템의 지능화가 이루어져야 한다.

### 참고문헌

- [1] G.H.M.B Carneiro, E.P.L Aude, H.Serdeira, J.T.CSilverira, M.F.Martins, *What n Internet Request Server Architecture for Telecommanding the CONTROLAB AGV through Real Time Data and Image?* IEEE, 1999.
- [2] Sebastien Grange, Terrence Fong and Charles Baur, "Effective Vehicle Teleoperation on the World Wide Web", *Proceeding of the 2000 IEEE International Conference on Robotics & Automation*, San Francisco, CA. April 2000.
- [3] Peter X.Liu, Max Q-H Meng, and Jason J.Gu, *Adaptive Scaling Control for the Internet-based Teleoperation?* *Proceedings of 2001 IEEE International Symposium on Computational Intelligence in Robotics and Automation* July 29 - August 1, 2001, Banff, Alberta, Canada.