

스트리밍 버퍼를 이용한 임베디드 시스템의 인터넷 원격제어

신완재*, 박장현#

Internet Teleoperation of an Embedded System using Streaming Buffer System

Wanjae Shin *, Jahng-Hyon Park#

ABSTRACT

Recently, necessity and usage of the teleoperation have been increased in various fields from industrial automation to home application. Also, the internet is considered as a strong candidate for the transmission media of signals. However, it has an irregular transmission time delay and causes critical problems such as instability and poor performance. This paper presents a practical internet teleoperation system with a streaming buffer system which makes a variable time delay fixed. Validity of the proposed system is demonstrated by implementing the embedded system on a HILS (hardware in the loop system) which models a two-wheel mobile robot.

Key Words: teleoperation(원격조종), streaming buffer(스트리밍 버퍼), time delay(시간지연), internet(인터넷), embedded system(임베디드 시스템)

기호설명

- n The size of buffer
- m The average of Round Trip Time during specified period
- σ The maximum of jitter
- δ The margin of streaming buffer for system stability
- V_o Output voltage of Pulse Width Modulation (PWM)
- T_{on} ON time of power supply of PWM

- T_{off} OFF time of power supply of PWM
- V_s Supply voltage
- V_R, V_L, V Velocity of right, left, center of Virtual Mobile Robot respectively
- $\omega_R, \omega_L, \omega$ Angular velocity of right, left, center of Virtual Mobile Robot

1. 서론

1940 년대에 처음 도입된 원격조종기술은 그 후 원자력 발전소에서의 방사능물질의 처리, 심해

접수일: 2004 년 2 월 25 일; 게재승인일: 2004 년 6 월 11 일

* 한양대학교 대학원 기계설계학과

교신저자: 한양대학교 공과대학 기계공학부

E-mail jpark@hanyang.ac.kr Tel. (02) 2290-0448

탐사, 우주공간에서의 작업, 미세수술 등 다양한 용도에 적용되고 있다¹. 한편, 인터넷이 새로운 통신수단으로 각광받기 시작하여 전화선을 이용하거나 직렬통신 등을 이용한 네트워크의 구축보다 더 빠르고 편리하게 원격지와 사용자간의 연결을 할 수 있게 되었다. 그러나 인터넷은 저가격과 쉬운 액세스, 빠른 전송 속도를 가지고 있지만 트래픽의 변화에 의한 전송속도의 변동으로 인하여 불규칙한 데이터 전송의 시간 지연이 발생하며 이는 원격조종에 사용시 사용자의 지시(command signal)의 시간에 대한 왜곡을 발생시켜 슬레이브 로봇(slave robot)이 사용자의 의도와 다른 행동을 하게 될 수 있으며 또한, 시스템의 불안정을 발생시킬 수 있다.

이의 대처방안으로 간단한 태스크들을 먼저 정의해 놓고 각각의 태스크를 수행하도록 명령어를 가하는 방식², 불규칙한 시간지연을 일종의 잡음(disturbance)로 간주하여 잡음에 강인한 구조로 제어기를 설계하는 슬라이딩 모드 제어³. 명령어의 수행여부를 확인 후 다음 명령어를 지시하는 이벤트기반 제어(event based control)^{4,5} 등이 제안되었다. 그러나 이들 제어방법은 시스템 전체의 대역폭이 줄어들거나, 시스템 제어기 구조가 복잡해지는 등의 단점들이 있다. 또한 필터를 채용한 파동변수법(wave variable using filter)⁶도 제안되었는데, 이 제어 방식은 매우 긴 시간의 시간지연의 경우에도 안정성을 확보할 수 있으나 마스터와 슬레이브 간의 힘의 지속적인 순환이 발생하는 단점이 있다.

본 논문에서는 사용자가 스트리밍 형태의 계속적인 명령신호를 내릴 때 각 신호의 시간지연의 불규칙성을 일정한 시간 지연으로 변환시키는 스트리밍 버퍼를 제안한다. 이는 간단한 구조를 가지고 있고 하위 제어기에 독립적이기 때문에 이미 구성되어있는 제어기에 연결이 가능하여 산업용 PC에서 임베디드 시스템까지 다양한 방면에 적용될 수 있다. 2장에서는 인터넷의 특징을 설명하고, 3장에서 스트리밍 버퍼를 설명하고, 이를 임베디드 시스템에 적용한 실험결과를 4장에서 보이고, 5장에서 결론을 설명하겠다.

2. 인터넷의 특징

월드와이드웹(WWW)의 등장으로 전세계의 인

터넷 사용자는 급격히 증가하였고, 이에 따라 설치, 유지비가 점차 감소하였다. 또한 이러한 인터넷은 MS 윈도우, 유닉스, BeOS 등의 운영체제와 PC, 셀룰러폰, PDA, 상업용 PC 등의 여러 하드웨어에 상관없이 사용가능하며 공장자동화에서 쓰이는 RS485 등의 직렬통신보다 빠른 전송속도를 가지고 있다. 그러나 이러한 데이터의 전송속도가 일정하지 않지 않기 때문에 각 패킷의 전송속도의 시간 지연이 불규칙하게 된다.(Fig.1) 따라서 이러한 신호를 원격조종시스템에 사용하게 되면 사용자가 원하지 않은 반응을 보일 수 있으며, 심지어 시스템의 불안정성을 초래하게 될 수도 있다.

Anderson 등⁷은 1/10 초의 극히 작은 시간 지연변화조차 시스템을 불안정하게 할 수 있다고 밝혔으며, Fig.1에서 보는 바와 같이 이러한 시간지연 변화는 인터넷을 사용할 시 피할 수 없는 현상이다. 이러한 시간지연변화의 양상은 해석적이지 않으며 통계론(stochastic)인 모습을 띄며, 이러한 인터넷의 시간지연 모델이 몇몇 연구자에 의해 연구되었다: 시간지연 역학의 모델을 웨이브렛으로 표현하거나⁸, 회귀신경망(Recurrent Neural Network)을 이용한 identification⁹ 등이 행하여졌다. 그러나 이러한 시간지연 시스템의 모델링이 가능하여도 원격조종시스템에서 명령어 전달의 신호 왜곡에 대처하는데 도움을 주지는 못한다.

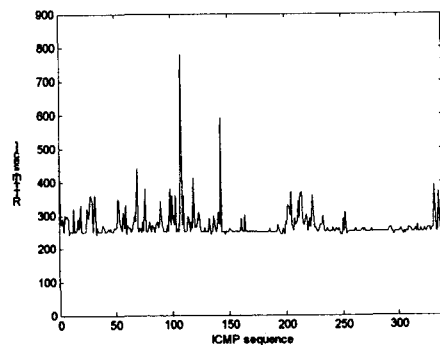


Fig. 1 Round Trip Time(RTT) using ICMP(Internet Control Message Protocol) between Seoul, Korea and Massachusetts, USA. ICMP Sequence time interval is 100msec and each packet size is 64bytes

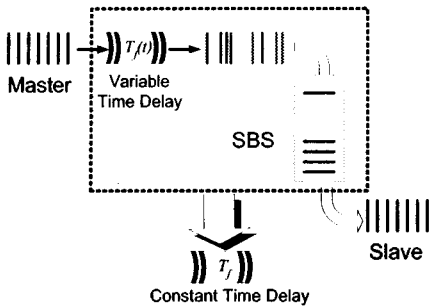


Fig. 2 Concept of Streaming Buffer System in case of unilateral teleoperation

한편, 신호전달의 규약에는 TCP/IP 와 UDP/IP 가 있다. 전자는 에러체크, 패킷의 자동 시퀀싱 등의 기능을 가지고 있으나 오버헤드가 있어서 후자에 비해 다소 늦는 전송속도를 가지고 있다. 반면 후자의 경우는 속도는 빠르나 패킷 손실의 가능성이 있으며 패킷시퀀스 순서의 뒤섞임이 발생할 여지가 있다. 원격조종시스템에서는 신호의 안전성이 중요하기 때문에 본 논문에서는 TCP/IP 를 이용한 신호전달 규약을 사용하였다.

3. 스트리밍 버퍼 시스템

스트리밍 버퍼시스템(Streaming Buffer System)의 개념은 들어오는 명령어를 버퍼링 하여 일정한 시간간격으로 내보내주는 일종의 FIFO(First In First Out) stack 이다.¹³ 이렇게 함으로써 다른 시간간격으로 들어온 신호를 같은 간격으로 내보내주게 한다.(Fig. 2) 이의 구조는 간단하기 때문에 여러 시스템에 부착하여 사용할 수 있다. 예를 들어 PC 의 경우에는 제어 알고리즘의 앞부분에 넣거나, 임베디드 시스템(Embedded System)의 경우 인터넷 연결부사이에 별도의 하드웨어로 구성할 수도 있다.

이의 구조는 Fig. 3 과 같이 되어 있으며, 크게 4 개의 부분인 버퍼매니저(buffer manager), 버퍼(buffer), 제어타imer(control timer), 신호재생기(signal reproducer)로 구성된다. 버퍼 매니저는 원격조종시스템을 시작할 때 버퍼의 크기를 정하여주며 마스터의 제어주기와 제어타imer의 제어주기를 같게 해준다. 마스터로부터 TCP/IP 를 통해 받은 신호는 스트리밍 버퍼에 쌓여 제어타imer의 클럭에 따라 신호재생기를 통해 먼저 들어온 순서대로 신호를 슬레이브의 하위 제어기로 보내게 된다.

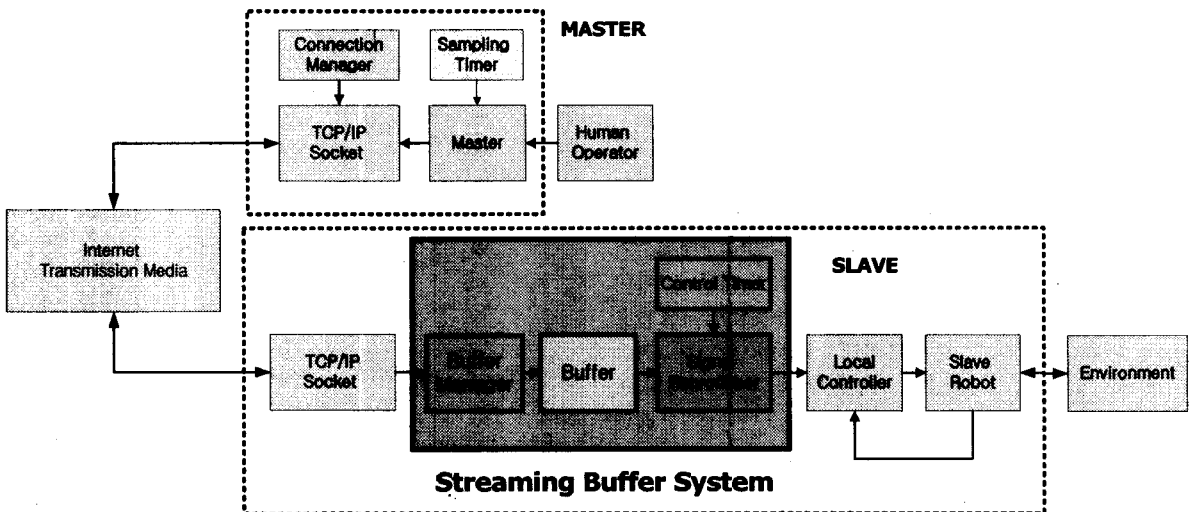


Fig. 3 The architecture of the entire system including the streaming buffer system

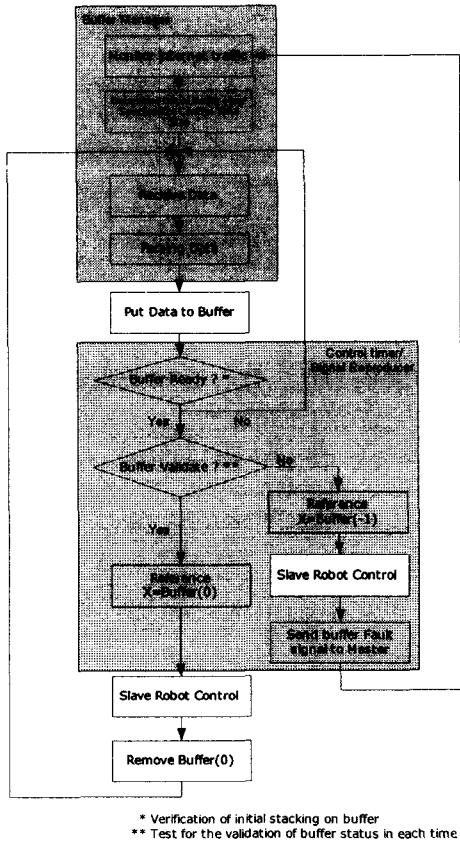


Fig. 4 Flowchart of Streaming Buffer System

버퍼매니저는 작업수행 전에 네트워크의 트래픽 상황을 고려해서 버퍼의 크기를 정해주는데 round trip time(RTT)과 그의 변화값인 jitter 를 이용하여 식 (1)과 같이 정해준다.

$$n = \text{ceil} \left(\frac{m + \sigma}{2T} \right) + \delta \quad (1)$$

여기서 $\text{ceil}(\cdot)$ 은 올림 함수, n 은 버퍼의 크기, T 는 마스터 측에서 보내는 신호의 주기 혹은 제어타이머의 주기, m 은 측정 시간 동안의 RTT 의 평균값, σ 는 jitter 의 최대값, δ 는 버퍼 크기 마진이다.

시간 지연이 버퍼의 용량보다 작다면 전체시스템은 안정하게 작업을 수행할 수 있다. 그러나 네트워크의 시간지연이 일시적으로 커져서 스트리

밍 버퍼에 더 이상 명령신호가 남아있지 않다면 버퍼 매니저는 버퍼 에러신호를 마스터에게 보내 주고, 하위제어기가 위치제어기일 경우는 버퍼에 남아있던 마지막 값을 전달하며, 속도제어기일 경우는 0 을 하위제어기에 전달하게 된다. 마스터에 있는 사용자는 이러한 에러신호를 받아들인 후 버퍼에 명령어를 다시 채우거나, 작업을 중단하는 등의 상황에 맞게 대처하게 된다.

4. 임베디드 시스템을 이용한 하드웨어-시뮬레이션 시스템 구성(Hardware In the Loop System: HILS)

4.1 임베디드 시스템(Embedded System)

과거에는 임베디드 시스템이라는 용어는 주로 미리 구성된 프로그램만을 수행하며 새로운 프로그램을 수용할 수 없어서 식기 세척기, 비행 네비게이션 시스템 등 특정작업만을 수행하도록 되어 있는 시스템을 일컬어서 일반적인 목적으로 쓰이는 PC 등과 뚜렷이 구별이 되었다.¹⁰ 그러나 하드웨어의 성능이 향상되고, 생산단가가 낮아짐에 따라 일반 목적 시스템과 임베디드 시스템과의 경계가 느슨해져 가는 추세이다.¹¹ 그러나 이러한 용어의 모호성에도 불구하고 임베디드 시스템은 주로 다음과 같은 특징을 갖는 시스템으로 통용된다. 즉, PC 보다 상대적으로 작은 크기이다, hard disk drive 같은 장치가 없다, 내장 프로그램이 한정적이다, 특정한 작업만을 위해 디자인되었다, 실시간 작업을 위한 운영체제가 내장되기도 한다 등이다.

4.2 HILS 실험

본 실험에서 사용된 임베디드 시스템은 슬레이브로 사용되었고, RISC 방식의 인텔 ARM 프로세서를 사용하여 DC 모터를 펄스폭 조정 방식(Pulse Width Modulation: PWM)으로 제어하였다. 인터넷 소켓과 스트리밍 버퍼를 구현하기 위해 리눅스를 운영체제로 하였으며, 타이머 인터럽트를 사용, PWM 의 펄스 폭을 제어하였다. PWM 의 평균 출력 전압은

$$V_o = \frac{T_{on} - T_{off}}{T_{on} + T_{off}} V_s \quad (2)$$

이며, 여기서 V_o 는 출력 전압, T_{on} 과 T_{off} 는 각각

전압공급의 ON/OFF 시간, V_s 는 공급 전압이다. 본 실험에서는 V_s 를 24 볼트로, 최소 T_{on}/T_{off} 주기는 2.56msec 로 하였다. 회전당 360 펄스를 내는 엔코더는 24bit 카운터를 이용, 입출력 포트 (General Purpose I/O: GPIO)를 통해 CPU 에 피드백 된다.¹²

실험 1 에서는 Table 1 에서와 같은 실험환경에서 마스터가 정현파 신호를 내보낼 때 물리적 거리 900km 떨어진 슬레이브가 수신하도록 하였다. 이때 버퍼 매니저는 네트워크 상황을 판단하여 버퍼크기를 5 로 선택하게 되었고, 이의 결과를 Fig. 5 에 보였다. Fig. 5a 는 기존의 방식을 이용한 결과이며 Fig. 5b 는 스트리밍 버퍼를 사용한 결과로서, 기존의 방식을 이용한 것보다 더 부드러운 반응을 보임을 알 수 있다.

Table 1 Environment of experiment 1

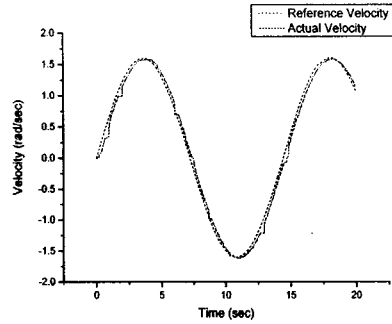
Distance	Control time step	Initial time delay	Calculated buffer size
about 900km	20msec	200msec	5

실험 2 에서는 가상 모바일 로봇(Virtual Mobile Robot: VMR)을 사용하였다. Fig. 6 에서와 같이 슬레이브에 2 개의 DC 모터를 부착하여 좌, 우 바퀴로 사용하였으며, 가상 모바일 로봇의 형상은 Fig. 7 과 같다. 마스터는 사용자가 2 자유도 햅틱 장치 (haptic device)를 이용하여 VMR 의 속도 명령신호를 주도록 하였고, 무선 랜을 사용하여 마스터와 슬레이브 간의 통신을 하도록 하였다.

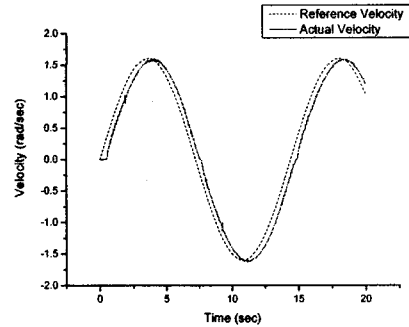
로봇에 고정된 좌표계에 대한 VMR 의 속도와 각속도는

$$\begin{aligned}
 V &= \frac{V_R + V_L}{2} = r \frac{\omega_R + \omega_L}{2} \\
 \omega &= \frac{V_R - V_L}{L} = r \frac{\omega_R - \omega_L}{L} \\
 \theta &= \int \omega dt
 \end{aligned}
 \tag{3}$$

가 되며, 여기서 V_R, V_L, V 는 각각 오른쪽, 왼쪽, 중심의 속도이다.



(a)



(b)

Fig. 5 The master's velocity command(dotted) and the motor(slave)'s velocity(solid) (a) when Streaming Buffer(SB) is not used. (b)SB is used

또한, $\omega_R, \omega_L, \omega$ 은 각각 오른쪽 바퀴, 왼쪽 바퀴, 중심의 각속도의 크기이다. 작업공간에 고정된 좌표계에 대한 VMR 의 속도와 각속도는 다음과 같다.

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \frac{r}{2} \cos \theta & \frac{r}{2} \cos \theta \\ \frac{r}{2} \sin \theta & \frac{r}{2} \sin \theta \\ 0 & -\frac{r}{L} \end{pmatrix} \begin{pmatrix} \omega_R \\ \omega_L \end{pmatrix}
 \tag{4}$$

여기서 $\dot{x}, \dot{y}, \dot{\theta}$ 는 각각 작업공간에 고정된 좌표계에 대하여 x, y 축 방향 선속도, z 축 방향 각속도

이다. ω_R, ω_L 은 2 개의 모터에 부착된 엔코더 신호로부터 얻어지는 위치신호를 수치미분하여 구해진다.

Table 2 와 같은 환경에서 15 초 동안의 VMR 의 위치 궤적을 Fig. 8 에 보였으며, Fig. 9 은 오른쪽과 왼쪽 모터의 속도를 나타내었다. 이때 버퍼의 크기는 네트워크 상황을 파악한 버퍼매니저에 의해 2 로 결정되었다.

Table 2 Environment of experiment 2

Distance	Control time step	Internet	Calculated buffer size
about 1km	20msec	Wireless LAN (802.11b, Wi-Fi)	2

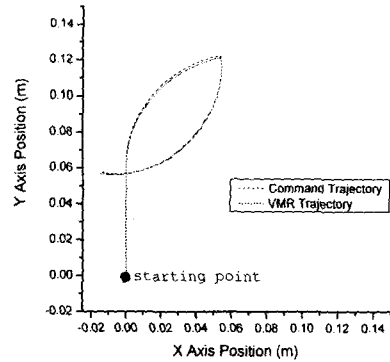


Fig. 8 Position Trajectory of Virtual Mobile Robot (dotted: command signal at the master, solid: actual position of VMR)

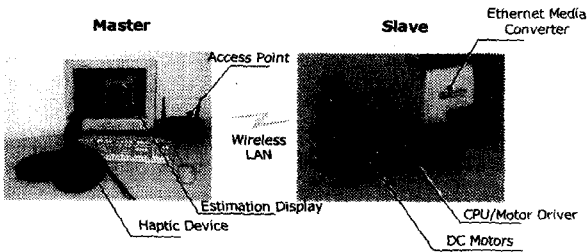


Fig. 6 System Configuration of the experiment 2

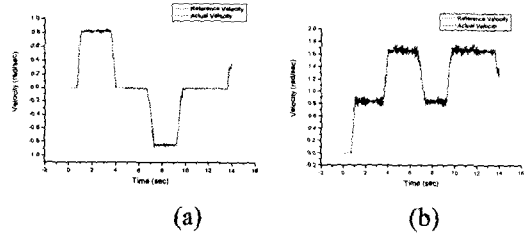


Fig. 9 Dotted: the master's velocity command, solid: the slave's velocity for (a) right motor velocity (b) left motor velocity

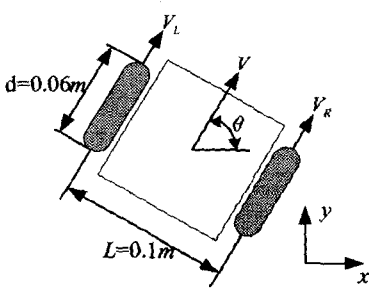


Fig. 7 Specification of Virtual Mobile Robot (VMR)

버퍼의 크기가 작은 경우, 마스터의 속도명령에 대해 적절히 추종하고 있음을 보여준다. Fig. 9b 에서 왼쪽, 오른쪽 모터의 채터링의 양상이 조금 다른 것은 두 개의 모터 사양이 다르기 때문에 발생하는 것으로 보인다.

한편 버퍼의 크기가 커지게 되면 속도 명령 신호의 경우 위치는 속도의 누적 값이기 때문에 VMR 실험의 경우 오히려 성능이 저하되는 경우가 발생할 수도 있다. 이 경우에는 위치신호를 명령신호로 가할 때 성능의 향상이 기대된다. 또한 실제 모바일 로봇을 구동할 때 발생하는 바퀴와 지면과의 미끄러짐 등을 고려할 때 위치 오차가 발생하기 때문에, 실제 시스템에서 적용하기 위해서는 각종 센서 등을 이용한 자기위치측정방법 (localization)이 필요할 것으로 보인다.

5. 결론

인터넷을 통한 원격조종의 단점으로 대표적인 것이 불규칙한 시간지연이 발생함으로써 시스템의 불안정성을 초래할 수 있다는 것이다. 본 논문은 이를 해소하기 위하여 불규칙한 시간지연을 고정된 값으로 변환시켜 주는 스트리밍 버퍼를 제안하였다. 스트리밍 버퍼는 슬레이브나 마스터의 제어기와 독립적이기 때문에 기존의 제어기에 추가하는 방식으로 쉽게 적용이 가능하다. 또한 네트워크의 상황을 고려하여 그 버퍼의 크기를 자동으로 결정하여 주기 때문에 수시로 변하는 네트워크의 상황에 맞는 대처를 할 수 있는 장점을 가지고 있다.

이의 유용성을 보이기 위해 본 논문에서는 DC 모터를 이용한 실험에서 기존의 방법보다 부드러운 반응을 보였으며, 2 개의 모터를 가지는 임베디드 시스템으로 하드웨어-시뮬레이션 장치(HILS)를 구성하여 가상 모바일 로봇의 속도제어를 수행하였다.

향후 연구 과제로 물체와 접촉하여 작업할 경우 명령신호와 실제 위치의 차이가 발생하는데 이에 대한 대처, 그리고 버퍼 크기의 변화에 의한 사용자의 시스템 운영에 대한 적응도 영향 등에 대한 연구가 필요할 것으로 보인다.

참고문헌

- Salcudean, S.E., hashtrudi-Zaad, K., Tafazoli, S., DiMaio, S.P., Reboulet, C., "Bilateral matched-impedance teleoperation with application to excavator control," IEEE Control System Magazine, Vol. 19, No. 6, pp. 29-37, Dec. 1999.
- Gorlberg, K., Gentner, S., Sutter, C., Wiegley, J., "The Mercury Project: A Feasible Study for Internet Robots," IEEE International Conference on Robotics and Automation(ICRA), pp. 1024-1028, May 1994.
- Park, Jong H., Cho, Hyun C., "Sliding Mode Control of Bilateral Teleoperation Systems with Force-Reflection on the Internet," Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2000), pp. 1187-1192, Nov. 2000.
- Xi, N., Tarn, T. J., "Action Synchronization and Control of Internet Based Telerobotic Systems," Proceedings of the 1999 IEEE International Conference On Robotics & Automation, pp. 219-224, 1999.
- Park, Jahng-Hyon, Park, Joonyoung, "Real Time Bilateral Control for Internet based Telerobotic system," IEEE/RSJ International Conference on Intelligent Robot and Systems(IROS 2003), pp. 1106-1110, Oct. 2003.
- Niemeyer, G., "Using Wave Variables in Time Delayed Force Reflecting Teleoperation," Ph.D. Thesis, MIT, 1996.
- Anderson, R., Spong, M., "Asymptotic stability for force reflecting teleoperators with time delay," Int. J. Robot. Res., Vol. 11, Apr. 1992.
- Riedi, R., Course, M., Ribeiro, V., Baraniuk, R., "A multifractal wavelet model with application to TCP network traffic," IEEE Trans. Inform. Theory(Special Issue on Multiscale Signal Analysis and Modeling), pp. 992-1018, Apr. 1999.
- Parlos, A.G., "Identification of the Internet End-to-end Delay Dynamics Using Multi-Step neuro-Predictors," IJCNN '02 Proc. of the 2002 International Joint Conference on Neural Networks, Vol. 3, 12-17, May 2002.
- Furber, S., "ARM system-on-chip architecture, 2nd Ed.," Addison-Wesley, pp. 293, 2000.
- Hollabaugh, Craig, "Embedded Linux: Hardware, Software, and Interfacing," Addison-Wesley, pp. 8-9, 2002.
- Intel Corporation, "Intel StrongARM SA-1100 Microprocessor Developer's Manual," Intel Corporation, pp. 9.1-9.10, June 2000.
- Park, Jahng-Hyon, Shin, Wanjae, "Internet Teleroperation of a Robot with Streaming Buffer System under Varying Time Delays," JSME International Journal Series C, Vol.46, No. 4 pp. 1604-1610, Dec. 2003.