

이더넷을 이용한 매니퓰레이터의 원격 유저 인터페이스 개발

이종수*, 류승엽**, 이기주***

(*충의대학교 전자전기 공학부 교수, **충의대학교 대학원 전기제어공학과 석사과정, ***충의대학교 대학원 석사과정)

The Development of a Remote User Interface for the Manipulator using the Ethernet

Lee, jong soo*, Ryoo, Sung yop**, Lee, ki ju***

(*School of Electronic & Electrical Eng., ** Graduate School of Electrical & Control Eng.)

Abstract - The operation of robot manipulators has a restriction that the operator must reside at the factory, where the manipulator is used. To overcome this restriction, we propose a remote control system using the internet, the system which runs on the Window 95 environment is composed of the remote client which transfers commands to the server which control and manage the manipulator in the factory. In the control of Hong-ik Direct Drive Arm, it is necessary to consider the complex nonlinear parameters causing the mutual interaction between joints, so we use two TMS320C31 DSP chips in the controller for the real time dynamic control algorithms. For the test of system integrity and the verification of the mathematical modeling, we apply CTM, PD and VSS control algorithms and the simulation results are satisfactory.

1. 서 론

기존의 로봇 매니퓰레이터의 운영은 로봇 매니퓰레이터가 사용되는 생산 현장에서 작업자가 상주하여 관리하는 제약이 있다. 이러한 제약을 극복하기 위해 최근 급속도로 보급된 인터넷을 활용하여 생산 현장 Server는 로봇 매니퓰레이터의 실제구동 및 서버를 수행하고, 작업자는 원격 Client에서 로봇 작업을 지시하고 관리하도록 하는 공장 자동화를 위한 구조를 제시한다.

본 논문에서는 윈도우 95환경에서 로봇 매니퓰레이터 제어기 및 서비스를 개발하였고, 적용대상으로는 2 자유도의 홍익 직접 구동 매니퓰레이터를 선정하였다. 홍익직접 구동 매니퓰레이터의 구동에는 관절간의 결합 토크 등의 상호 간섭의 영향을 고려하는 동적 제어 알고리즘이 필요하고, 이러한 동적 제어알고리즘의 복잡한 계산을 실시간으로 처리하기 위해 많은 양의 연산을 단 시간 내에 처리할 수 있는 Texas Instruments 사의 TMS320C31 DSP 칩 2개를 사용하여 제어기를 설계하였다. 제어기의 성능 실험과 수학적 모델링의 검증을 위해 CTM 제어, Feedforward 보상 PD의 채환제어, 가변 구조 제어가 성공적으로 수행됨을 보였다.

2. 본 론

2.1 전체 시스템의 구성

전체 시스템의 구성은 작업을 지시하는 Client와 실제 구동을 담당하는 Server 부분으로 나눌 수 있다. Server는 다시 Host computer와 제어기 부분으로 나누어진다. 그림 1은 시스템의 구성을 개괄적으로 나타낸 것이다.

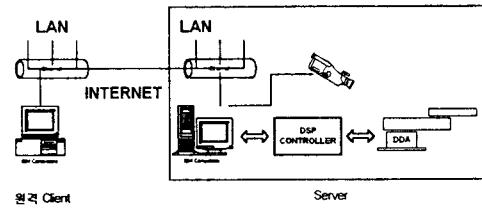


그림 1. 전체 시스템 구조

2.2 Client 시스템의 구성

클라이언트는 소프트웨어인 통합 환경 Tool과 교시용 하드웨어로 구성되어 있다. 통합 환경 Tool은 사용자가 원격 지의 매니퓰레이터의 구동하기 위한 데이터를 작성하고 실행하는데 필요한 모든 기능을 제공하기 위한 것으로 편집, 교시, 컴파일, 시뮬레이션, 통신의 5가지 기능으로 구성되어 있다.

2.2.1 편집 기능

편집 기능은 매니퓰레이터가 수행할 일련의 작업(job file)을 작성하고 수정하도록 하기 위한 것으로 사용자는 이동명령과 제어명령으로 이루어진 로봇 프로그래밍 언어로서 이를 작성하게 된다. [6] job file은 줄 단위로 이루어져 있으며 이동 또는 제어 명령어와 위치, 이동시간, 경로 계획 순서로 입력하게 되어 있다. 명령어와 경로 계획의 입력은 팝업(Pop-up) 메뉴방식에 의해 마우스로 간단히 선택할 수 있다.

2.2.2 교시 기능

교시 기능은 매니퓰레이터를 3차원으로 모델링한 그래픽 인터페이스를 이용하거나 외부 교시용 하드웨어 인터페이스를 이용하여 매니퓰레이터의 각 관절의 위치를 지정하고 입력하기 위한 것이다. 교시기능은 job file의 작성 시 이용되거나 좌표 테이블에 저장하는 위치 변수 정의에 사용된다.

2.2.3 컴파일 기능

편집기를 통해 작성된 job file은 그 자체가 서버에 전송되는 것이 아니라 원격 지의 매니퓰레이터를 경로를 생성하는데 필요한 계수로 변환되어 전송된다. 컴파일 기능은 이렇게 로봇 프로그래밍 언어로 작성된 job file을 계수로 변화시켜 주기 위한 것이다.

2.2.4 시뮬레이션 기능

시뮬레이션 기능은 작성된 job file에 의해 예상되는 원격 지의 매니퓰레이터 동작을 3차원 그래픽을 이용하

여 시각적으로 시뮬레이션 하는 기능으로 한번 전송하면 이를 되돌리기 어려운 네트워크의 특성을 보완하기 위한 것이다. 사용자는 전송 전에 작성한 job file을 검증할 수 있기 때문에 잘못된 job file의 전송에 의해 발생되는 오류를 사전에 방지할 수 있다.

2.2.5 통신 기능

통신 기능은 원격 지의 서버에 접속한 후 네트워크를 통해 서버로 필요한 데이터를 전송하거나 서버로부터 수신하는 기능이다. 이를 통해 job file의 컴파일 후 생성된 계수를 적용할 제어 알고리즘을 선택하여 함께 서버로 전송하고 서버로부터는 원격 지의 작업장소를 모니터링 할 수 있도록 서버에 부착된 CCD 카메라를 통해 전송되는 화상 데이터를 수신한다.

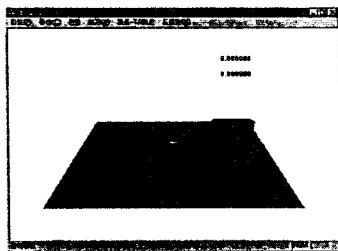


그림 2. 통합 환경 Tool

2.3 Server 프로그램

Window 95환경의 Host PC와 제어기의 인터페이스를 위해 여러 개의 VM(Virtual Machine) 각각에 대해 동일한 가상적 하드웨어를 제공하는 실행 루틴의 집합인 VxD 프로그래밍이 필수적이다. 하드웨어 제어에 있어서 기본적인 프로그램 요소 I/O 동작, 인터럽트 핸들링, 특정 메모리 접근, DMA를 Window 95에서 구현하기 위해서는 VMM, VPICD, VDMA에서 제공하는 Service 루틴을 사용하였다.

2.4 Server의 제어기 구성

2.4.1 Control Board의 구성

본 논문에서는 Server측에서 제어와 경로 계획을 실시간으로 처리하기 위해 2개의 TMS320C31을 이용하여 제어기를 구성하였다. 제어기의 구조와 데이터의 흐름은 다음 그림과 같다.

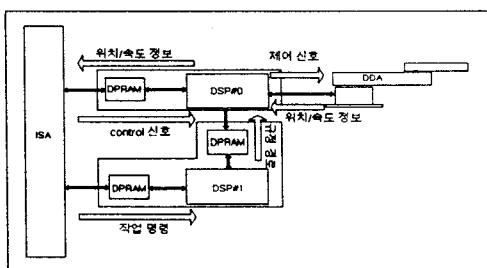


그림 3. 제어기의 전체 구성

2.4.2 I/O Board의 구성

I/O Board는 모터 드라이브와 결합하여 제어기에서 계산한 토오크 값과 모터에서 출력되는 엔코딩 신호를 제어기와 모터드라이브 사이에서 중계해주는 역할을 한다.

3. 실험 및 검토

3.1 Client에서의 경로 계획

다음 그림은 각각 서버에 전송된 계수를 이용하여 서버 프로그램에서 제 1관절과 제 2관절의 경로를 그래프로 시뮬레이션 한 것이다. 계수는 4-3-4 경로 계획에 의해 산출되었다. 제 1관절은 시작 위치를 0° , 목표 위치를 90° 로 하였고, 제 2관절의 시작 위치는 0° , 목표 위치는 120° 로 설정하였다.

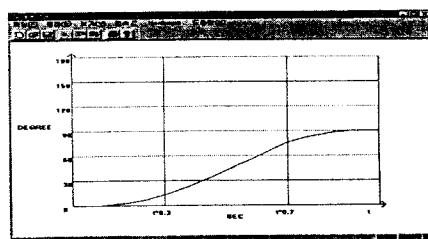
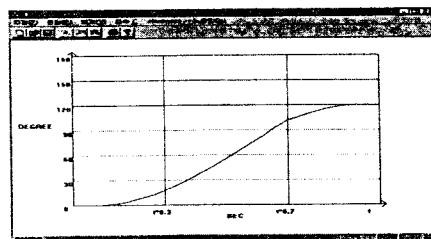


그림 4. 경로 시뮬레이션

Client가 Server로 로봇의 작업 위치 정보를 전송하면 Server는 Dual-Port RAM을 통하여 DSP#1에 경로 계획에 필요한 계수를 전달하고 DSP#1은 이 정보로 1ms마다 경로를 발생시켜서 DSP#0에 전달한다. DSP#0은 전달된 계수를 이용하여 독립적인 로봇제어 알고리즘을 실행하며 결과를 Dual-Port RAM을 통하여 Host Computer에 전달한다. 제어기의 성능 실험과 수학적 모델링의 검증을 위해 CTM 제어, 가변 구조 제어, Feedforward 보상의 PD 케환 제어 방식으로 흥의 직접 구동 팔을 실험하였다. 다음 그림은 세 가지 알고리즘을 이용하여 실험한 결과를 error값으로 나타낸 자료다.

3.2 제어 알고리즘

3.2.1 CTM

원하는 규정 경로가 발생될 경우 이를 $q_d(t)$ 라고 하면

$$\begin{aligned} e(t) &= q_d(t) - q(t) : \text{Tracking Error} \\ \dot{e}(t) &= \dot{q}_d(t) - \dot{q}(t) \\ \ddot{e}(t) &= \ddot{q}_d(t) - \ddot{q}(t) \end{aligned}$$

CTM에서의 제어 입력 $\tau(t)$ 는 다음과 같다.

$$\tau(t) = M(q)(\ddot{q}_d + K_p e + K_v \dot{e}) + N(q, \dot{q})$$

$$K_p = 5000, K_v = 150$$

3.2.2 Feedforward 보상에 의한 PD 제어

Feedforward 보상과 PD 케환제어를 이용한 알고리즘에서 모델을 이용한 입력 토오크는

$$M(q_d)\ddot{q}_d + N(q_d, \dot{q}_d) + \tau_{fd}(e, \dot{e}) = \tau$$

이에 의한 오차 방정식은

$$\begin{aligned} \tau_{fd}(e, \dot{e}) &= M(q)\dot{q} + N(q, q) \\ &\quad - M(q_d)\ddot{q}_d - N(q_d, \dot{q}_d) \end{aligned}$$

케환 제어 성분은

$$\tau_{fd} = K_p e + K_v \dot{e}$$

$$K_p = 5000, K_v = 150$$

로 설정하였다.

3.2.3 가변 구조 제어

입력 토오크는

$$M(q_d)\ddot{q}_d + N(q_d, \dot{q}_d) + \tau_s(e, \dot{e}) = \tau$$

이에 의한 오차 방정식은

$$\begin{aligned} \tau_s(e, \dot{e}) &= M(q)\dot{q} + N(q, q) \\ &\quad - M(q_d)\ddot{q}_d - N(q_d, \dot{q}_d) \end{aligned}$$

이다.

슬라이딩 모드에 의한 제어 성분은

$$\tau_s = K_1 \frac{s}{|s|} + K_2 s, s = \dot{e} + K_0 e$$

아래 관절 $K_0 = 800, K_1 = 30, K_s = 80$

윗 관절 $K_0 = 700, K_1 = 10.5, K_s = 30.5$

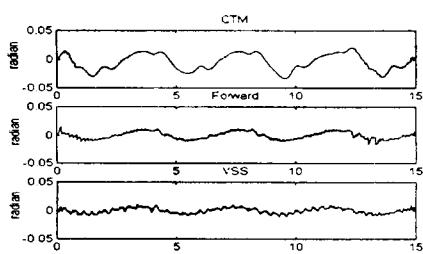


그림 5. 무부하 실험시의 링크1의 위치 에러

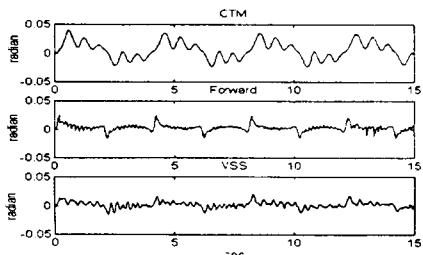


그림 6. 무부하 실험시의 링크2의 위치 에러

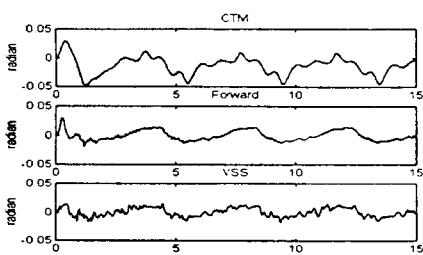


그림 7. 7Kg 부하 실험 시 링크 1의 위치에러

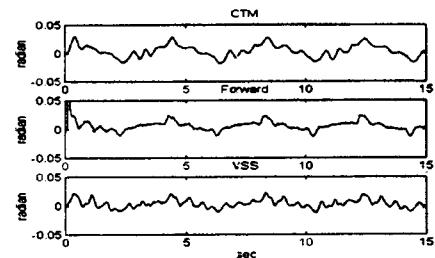


그림 8. 7Kg 부하 실험 시 링크 2의 위치 에러

부하	무부하		부하=7kg	
	관절1	관절2	관절1	관절2
CTM	0.0152	0.0113	0.0159	0.0204
PD	0.0065	0.0086	0.0067	0.0103
VSS	0.0046	0.0075	0.0057	0.0080

표 1 rms 위치 오차 [단위 : rad]

4. 결 론

실험 결과, 부하에 관계없이 가변 구조 제어 방식이 가장 오차가 적었고 다음으로 Feedforward 보상 PD 케환 제어, CTM 순서로 오차가 적었다. 실험을 통하여 제한적인 세 알고리즘 중 홍익의 직접 구동팔을 동작시키는데 가변구조 제어 방식이 가장 효과적인 알고리즘임을 알 수 있었다.

(참 고 문 헌)

- [1] K.S. Fu, R.C. Gonzalez, C.S.G. Lee, ROBOTIS Control,Sensing,Vision, andIntelligence,McGRAW-HILL, 1987.
- [2] J.J. Craig, Introduction to ROBOTICS, Addison Wesley, 1989.
- [3] C. Wahnun, 3-D Graphics Programming with OpenGL, QUE, 1997.
- [4] R. Davis, Win32 Network Programming, Addison Wesley, 1996.
- [5] D. Drascic, P. Milgram, A. Rastogi, "Telerobotic Control with Stereoscopic Augmented Reality", SPIE vol. 26530, pp. 135-146, San Jose, 1996.
- [6] 김성훈, "MC68000을 이용한 홍익 직접 구동팔의 동적 제어에 관한 연구", 석사학위 논문, 홍익대학교, 1993.
- [7] 우상수, OpenGL 프로그래밍, 사이버 출판사, 1997.
- [8] 이종수, "로보트의 하이브리드 제어기 설계에 관한 연구" 박사학위 논문, 서울대학교, 1987.