

네트워크를 이용한 AM1 로봇의 원격 동적 제어

Remote Dynamic Control of AM1 Robot Using Network

김성일¹, 배길호², 김원일³, 한성현³

1. 경남대학교 대학원 기계설계학과 (TEL: 055-249-2590, FAX: 055-249-2617)
2. 미래산업(주) (TEL : 031-780-1410, FAX : 031-780-6910)
3. 경남대학교 기계자동화공학부 (TEL: 055-249-2624, FAX: 055-249-2617)

Abstract

In this paper, we propose a remote controller for robot manipulator using local area network(LAN) and internet. To do this, we develop a server-client system as used in the network field. The client system is in any computer in remote place for the user to log-in the server and manage the remote factory. the server system is a computer which controls the manipulator and waits for a access from client.

The server system consists of several control algorithms which is needed to drive the manipulator and networking system to transfer images that shows states of the work place, and to receive a Tmp data to run the manipulator. The client system consists of 3D(dimension) graphic user interface for teaching and off-line task like simulation, external hardware interface which makes it easier for the user to teach. Using this server-client system, the user who is on remote place can edit the work schedule of manipulator, then run the machine after it is transferred and monitor the results of the task.

1. 서론

현재 산업용 매니플레이터는 자동화 산업을 이끌어 나가는데 있어서 중요한 역할을 담당하고, 공장 자동화의 확산에 따라 유연성을 가지고 다양한 분야에서 널리 사용되고 있다. 하지만, 기존 매니플레이터의 경우 작업자가 매니플레이터가 사용되는 작업장에 상주하여 작업하는 형태로 되어 있기 때문에 장소를 달리하는 경우는 장소의 수에 따라 인원이 배치되어야 하는 비효율적인 측면과 인간을 대신하여 위험하거나 어려운 일을 대신 해야 함에도 인간이 같은 작업환경에 있어야 한다는 문제점이 대두되고 있다.

사용자가 원거리에 있는 매니플레이터를 조작하는 개념의 Telerobotics는 이러한 문제점을 해결할 수 있는 방법으로서 여러 가지 분야에서 많은 연구 및 활용사례가 발표되어 왔다. 하지만 기존의 Telerobotics의 가장 큰 문제점은 원격제어를 위해 원격지와 매니플레이터간을 연결해 주는 제어용의 별도의 전용라인을 필요로 하였고 이러한 전용라인의 설치는 거리에 비례하여 비용 및 시공하는데 어려움이 증가하기 때문에 극히 제한된 거리에서만 그 구현이 가능한 실정이다.

하지만 네트워크를 이용한 로봇의 원격제어는 기존의 보급되어 있는 디지털 네트워크를 이용하기 때문에 별도의 비용을 들이지 않고 원격지의 로봇을 제어할 수 있는 환경을 구축할 수가 있다. 특히, 이미 널리 보급되어 있는 인터넷과 연

제시키는 경우 인터넷이 사용 가능한 곳이면 지구 어느 곳이라도 원격 제어가 가능하기 때문에 기존의 문제가 되었던 거리에 따른 제약 문제를 해결 할 수 있다.

디지털 네트워크 기술을 활용한 원격제어의 개발은 작업자가 실제 로봇과의 거리에 관계없이 로봇을 제어할 수 있는 환경을 구현하여 보다 능률적인 공장 자동화를 비교적 저렴한 비용으로 구성하여 생산 현장의 효율성을 증가시킬 수 있는 기반을 제공함과 동시에 작업자를 유해환경이나 위험한 작업에서 해방시켜주는 무인시스템의 개발 및 확대에 지대한 역할을 할 것이다.

2. 제안된 시스템의 구성

2.1 시스템의 전체적 개요

원격제어 시스템은 그림1과 같이 크게 Client와 Server로 나눌 수 있다. 이는 네트워크에서의 개념을 도입한 것으로 실제 매니플레이터와 연결되어 있는 컴퓨터를 Server 그리고 이에 접속하여 원격제어를 실현한 컴퓨터를 Client로 한 것이다.

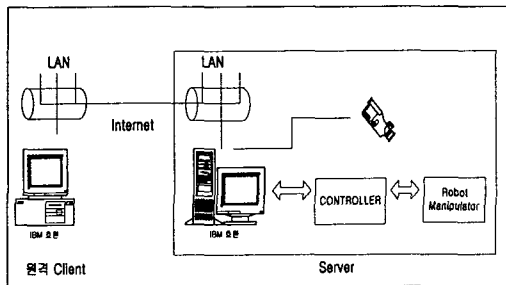


그림1. 전체시스템 구성도

2.2 Client 시스템의 구성

클라이언트는 소프트웨어인 통합 환경 툴과 교시용 하드웨어로 구성되어 있다. 통합 환경 툴은 사용자가 원격지의 매니플레이터를 구동하기 위한 데이터를 작성하고 실행하는데 필요한 모든 기능을 제공하기 위한 것으로 편집, 교시, 컴파일, 시뮬레이션, 통신의 기능으로 구성되어 있다. 교시용 하드웨어는 원격지의 매니플레이터를 모형화시켜 사용자에게 좀 더 편리하고 실제적인

교시 기능을 제공하기 위한 것으로 각 관절의 위치를 표현하기 위한 2개의 다회전 Potentiometer 와 A/D Convert 그리고 Interface Card로 이루어져 있다.

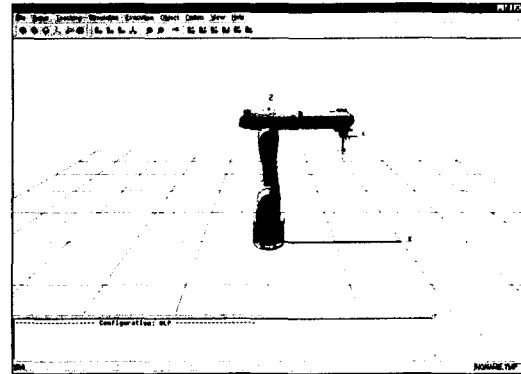


그림2. 통합환경툴의 실행모습

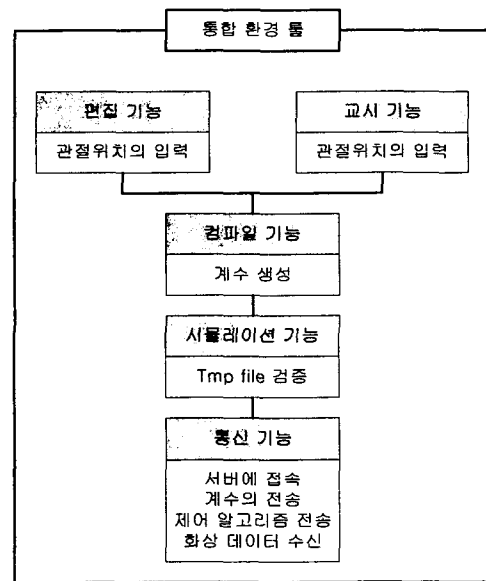


그림3. 시스템 흐름도

2.2.1 편집기능

편집 기능은 매니플레이터가 수행할 일련의 작업(Tmp file)을 작성하고 수정하도록 하기 위한 것으로 사용자는 이동명령과 제어명령으로 이루어진 로봇 프로그래밍 언어로서 이를 작성하게 된다.[6] Tmp file은 줄 단위로 이루어져 있으며

이동 또는 제어 명령어와 위치, 이동시간, 경로 계획순으로 입력하게 되어있다. 명령어와 경로 계획의 입력은 풀다운 메뉴방식에 의해 마우스로 간단히 선택할 수 있다. 편집기는 그림 4와 같으며 편집기능외에 간단한 로봇 프로그래밍 언어를 개별로 실행할 수 있는 버튼메뉴가 있다. 편집기 상에서는 각 관절의 변위값과 이에 따른 End-effector의 직각 좌표상에서의 값이 디스플레이 된다. 이밖에 좀 더 세밀하게 관절의 위치 값을 증가 또는 감소시킬 수 있는 버튼 메뉴가 있다.

2.2.2 교시기능

교시 기능은 매니플레이터를 3차원으로 모델링한 그래픽 인터페이스를 이용하거나 외부 교시용 하드웨어 인터페이스를 이용하여 매니플레이터의 각 관절의 위치를 지정하고 입력하기 위한 것이다. 교시는 교시메뉴에서 대화 상자를 통해 교시 정보를 저장할 교시파일을 만들고, 여러 교시정보를 입력한 후, 교시 파일을 저장함으로써 이루어진다. 그림5는 교시기능으로 관절을 변화시킨 모습이다.

2.2.3 컴파일 기능

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

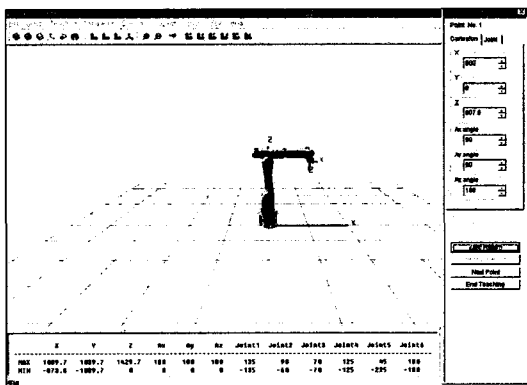


그림4. 편집용 대화상자 실행

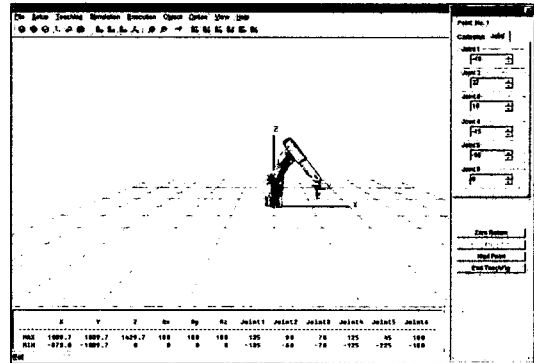


그림5. 교시기능으로 관절을 변화시킨 모습

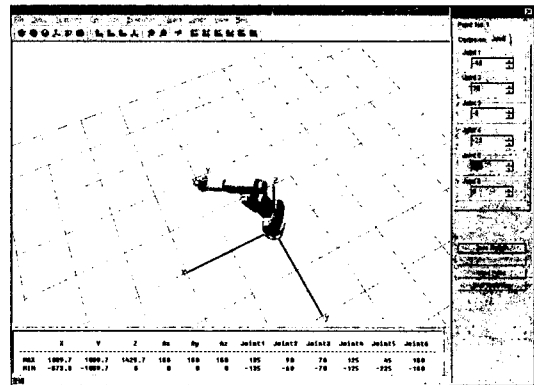


그림6. View point가 변경된 모습

2.2.4 시뮬레이션 기능

시뮬레이션 기능은 작성된 Tmp file에 의해 예상되는 원격지의 매니플레이터 동작을 3차원 그래픽을 이용하여 시각적으로 시뮬레이션 하는 기능으로 이는 시뮬레이션 궤적 확인에 의해 제어 추종 성능 평가가 가능하고 한번 전송하면 이를 되돌리기 어려운 네트워크의 특성을 보완하기 위한 것이다. 사용자는 전송전 작성한 Tmp file을 시뮬레이션 기능으로 검증할 수 있기 때문에 잘못된 Tmp file의 전송에 의해 발생하는 오류를 사전에 방지할 수 있다.

2.2.5 통신기능

통신 기능은 원격지의 서버에 접속한 후 네트

워크를 통해 서버로 필요한 데이터를 전송하거나 서버로부터 수신하는 기능이다. 이를 통해 Tmp file의 컴파일 후 생성된 계수를 적용할 제어 알고리즘을 선택하여 함께 서버로 전송하고 서버로부터는 원격지의 작업장소를 모니터링 할 수 있도록 서버에 부착된 CCD카메라를 통해 전송되는 화상 데이터를 수신한다.

2.2 Server 시스템의 구성

서버 시스템은 CCD카메라를 통해 들어온 화상을 capture하는 기능 및 클라이언트와 같이 데이터를 전송 및 수신하는 기능 그리고 매니플레이터를 구동하기 위한 하드웨어를 초기화하는 기능을 가지고 있다. 윈도우즈98의 특성상 하드웨어와의 인터페이스는 VxD라는 가상 드라이버를 통해 이루어진다. 구동을 위한 하드웨어는 드라이버와 드라이버에 입력할 수치를 연산할 DSP(Digital Signal Processing)보드로 구성되어 있다.

2.3 프로그램 개발 환경

시스템은 마이크로소프트사의 비주얼 C++ ver 5.0에 의해 개발되었다. 비주얼 C++은 윈도우 프로그램을 쉽고 구조적으로 제작할 수 있는 MFC와 각종 Wizard 즉, 기본적인 source를 자동으로 코딩하여 주는 기능을 제공하므로, 개발된 프로그램은 차후 다른 여러 사용자들에 의해서 간단히 확장되거나 이식될 수 있다. 서버와 클라이언트간의 네트워크 통신 기능은 Winsock 클래스를 이용하였고 3차원 그래픽은 OpenGL이라는 3차원 그래픽 전용 라이브러리를 이용해 구현하였다.

2.3.1 Winsock 클래스

Winsock은 windows에서 TCP/IP(Transmission Control Protocol/Internet Protocol) 응용 프로그램을 만들 때 사용하는 API(Application Programming Interface)의 집합이다. TCP/IP는 프로토콜로서 에러 없는 데이터의 전송과 에러의 검출, 복구 그리고 순차적인 전달 및 전송 도중 packet이 분실되거나 파손되는 경우 문제 발생을 알리고 데이터를 알맞게 재전송 하는 역할을 한다. Winsock은 일종의 함수 프로토콜타입을 가지고 있는 헤더 파일이며 이 헤더 파일에는 Winsock

사양에서 정의되어 있는 모든 함수들이 선언되어 있다.[4]

2.3.1 OpenGL Library

OpenGL은 새로운 산업용 3D 그래픽 표준으로 등장하고 있는 Library이다[7]. 윈도우 상에서 빠른 3D 그래픽 능력과 간단한 프로그래밍 명령어로 프로그램 개발에 유용하게 활용되고 있다.

OpenGL은 2차원의, 원근 좌표계로 표현시 확대 및 축소, 물체의 질감이나 조명효과, 애니메이션과 같은 3차원 그래픽의 난점을 해결하기 위한 것으로 실리콘 그래픽스사가 워크스테이션에 사용하던 IRIS GL을 운영체계에 관계없이 사용할 수 있도록 여러 회사와 컨소시엄을 구성하여 만든 표준 3차원 그래픽 라이브러리이다. 표준이 확정된 후 여러 가지 기종으로 포팅(porting)되어 SUN, WINDOW NT, WINDOW 98에서 사용할 수 있게 되었다.[3,5]

3. 적용 결과 및 검토

이상에서 언급한 원격 제어 환경하에서 그림 3의 흐름도와 같이 Tmp file을 작성하고 컴파일한 뒤 생성된 계수를 시뮬레이션을 거쳐 서버에 전송하였다. 그림 8은 각각 서버에 전송된 계수를 이용하여 서버 프로그램에서 첫 번째 관절에서부터 여섯 번째 관절의 경로를 그래프로 시뮬레이션 한 것이다. 첫 번째 관절부터 여섯 번째 관절 모두 시작위치를 0°로 하고 목표위치를 20°로 한 것이다.

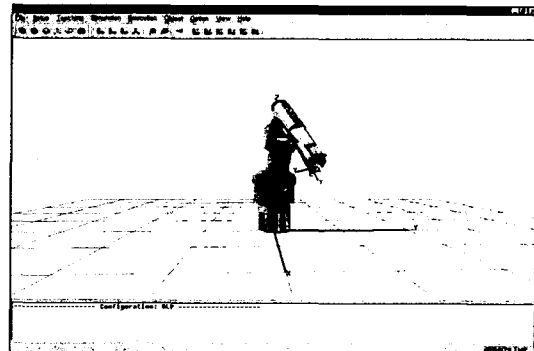


그림7. 6개의 각 관절이 20° 변화된 모습

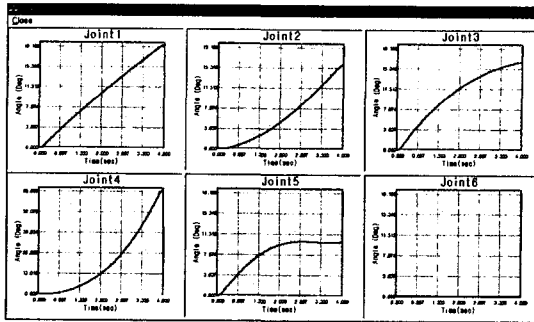


그림8. 각 관절의 경로

4. 결론

본 논문에서는 윈도우즈 98 환경하에서 네트워크를 이용하여 서버-클라이언트 개념을 도입해 원격지의 매니플레이터를 제어할 제반 시스템을 개발하였다. 사용자는 이 시스템을 이용 클라이언트 시스템에서 필요한 Tmp file을 작성하고 시뮬레이션을 통해 검증한 뒤 이를 서버 시스템으로 전송하여 매니플레이터를 안전하고 정확하게 구동할 수 있다.

참고문헌

- [1] K.S. Fu, R.C Gonzalez, C.S.G Lee, ROBOTICS Control, Sensing, Vision, and Intelligence, McGRAW-HILL, 1987.
- [2] J.J. Craig, Introduction to ROBOTICS, Addison Wesley, 1989.
- [3] C. Wahnun, 3-D Graphics Programming with Open GL, 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 직접 구동팔의 동적제어에 관한 연구", 석사학위 논문, 홍익대학교, 1993

- [7] Ron Fosner, OpenGL Programming for Windows 98 and windows NT, Addison - Wesley Developers Press, 1997