

인터넷 기반 가상실험실의 구현

Implementation of Virtual Laboratory Based on the Internet

주영훈* · 김문환** · 이호재** · 박진배**

Young Hoon Joo · Moon Hwan Kim · Ho Jae Lee · Jin Bae Park

*군산대학교 전자정보공학부, **연세대학교 전기전자공학과

요 약

많은 공학 연구에서 실제 시스템을 활용한 검증은 매우 중요하다. 특히, 제어공학의 경우 개발된 새로운 이론을 적용 및 검증할 수 있는 실험장비의 확보는 필수적이다. 그러나, 현실적으로 적절한 실험장비를 확보하는 것은 비경제적이다. 인터넷을 활용하여 실험장비의 원격실험을 가능케 한 가상실험실은 제어공학 연구에 유용하다. 인터넷의 보급과 함께 가상실험실에 대한 연구가 활발히 진행되고 있으나, 대부분의 경우가 특정 시스템에 한정된 가상 실험실의 구축에 한정되어 있다. 본 논문에서는 기존 연구와 달리 다양한 실험 장비에 적용 가능하며 다양한 제어 기법을 제공하는 가상실험실의 구현을 제안한다. 제안된 가상실험실은 서버-클라이언트 구조를 가지며 다중 제어 인터페이스로 구현된다.

Abstract

This paper concerns the establishment of the Internet-based virtual laboratory (VL). In control engineering, it is required to evaluate the feasibility of a newly developed controller design technique by applying to a physical system. However, it is inefficient to make or build such a experimental apparatus in all research activities. A possible remedy is to share such a apparatus spatially via the Internet. We set up techniques for the remote-control of various experimental apparatuses based on the Internet. The proposed VL forms a server-client structure and is implemented in multi-control interfaces.

Key Words : 가상실험실, 인터넷, 원격제어, 서버-클라이언트, 보안인증

1. 서 론

제어 공학 분야의 연구를 원활히 수행하기 위해서는 고급 이론의 개발과 개발된 이론의 실제 시스템에 대한 적용을 모두 효율적으로 연구 및 검증할 수 있어야 한다. 그러나, 한정된 재원을 갖는 연구기관이나 학교에서 적절한 실험 기자재를 적시에 공급하는 것은 매우 어려운 일이다. 현재 폭발적으로 보급되고 있는 인터넷은 그 광범위한 응용분야로 공학 교육 및 연구의 훌륭한 기반역할을 수행할 것으로 예상되고 있다. 본 논문에서는 이러한 요구 사항과 현재 보급되고 있는 인터넷의 효용성을 결합한 인터넷 기반 가상 실험실을 구축을 논의한다.

인터넷 기반 가상 실험실은 현재 국내외적으로 연구가 진행되고 있으나보다 다양한 환경에서의 활용을 위한 많은 연구가 필요하다. 또한 인터넷 기반 제어는 그 구현의 유연성과 다양함으로 인해 많은 종류의 구축 방법이 제안되고 있다. 1990년대 초의 인터넷의 보급 확산을 시발점으로 삼아 많은 이론적인 발전과 실제 구현 사례가 발표되었다 [1-13]. Weaver[1]와 Piguat[9]는 자바(JAVA) 프로그래밍 언어를 사용하여 관측 시스템의 상황을 원격지로 전달하고 간단한

명령을 전송하는 시스템을 개발하였다. Leleve [5]는 인터넷을 통해 원격 조작되는 로봇 시스템의 전체적인 모델링을 연구, 이를 모의실험으로 검증하였고 Young [10]은 인터넷을 통해 실시간으로 원격지의 제어 대상 시스템의 상황을 모니터링 하는 방법을 제안하였다. 그러나, 대부분의 연구결과는 특정한 응용분야를 대상으로 구축되어 일반적인 제어 이론의 평가 및 제어공학 교육에의 활용이 용이치 않다. 더욱이 기존 연구결과는 제어 시스템 전반의 상황이 바뀌는 경우 이에 유연하게 대처하는 것이 용이하지 않은 실정이다.

간단한 시스템의 인터넷 기반 가상실험실, 또는 교육용 가상실험실의 구축에는 특정한 제어기법을 선택하여 시스템을 구축하는 것이 편리하나, 복잡한 시스템으로 구축된 연구용 가상실험실의 경우는 여러 제어 방법을 선택적으로 유연하게 적용할 수 있는 시스템 구성이 필요하다. 더욱이 제어 대상 플랜트는 지금까지 이러한 인터넷에 어울리는 개방형으로 설계되지 않았으므로 이를 인터넷에 개방성을 지니며 연결할 수 있도록 하는 기술의 개발이 필요하다.

본 논문에서는 보다 제어 대상 플랜트에 유연하고 다양한 제어 형태를 제공하는 가상실험실의 구축을 제안한다. 이를 위하여 전체적인 가상실험실은 서버-클라이언트 구조로 구성된다. 서버 프로그램의 경우 다양한 방법으로 제어 방법과 제어 결과를 표현 할 수 있도록 자바를 이용하여 전체 응용 프로그램을 작성한다. 제어 대상 플랜트를 위한 응용프로그램의 경우 플랜트의 유연성을 확보하기 위하여 시스템 제어 부분, 통신 부분, 제어 신호 발생 부분으로 나누

접수일자 : 2003년 5월 4일

완료일자 : 2003년 8월 7일

본 논문은 한국과학재단(R-02-2000-00252)의 지원에 의하여 연구되었음.

어서 구성되며, 하드웨어 인식률이 가장 좋은 C++ 을 이용하여 구현된다.

2. 사용자 인터페이스

2.1 연결 제어 인터페이스

연결 제어 인터페이스(Interface)는 플랜트 제어 프로그램과 원격 제어 프로그램의 사이의 통신을 제어하기 위한 인터페이스를 제공한다. 기본적으로 연결과 연결 종료의 두 가지 기능을 제공하며 연결과 연결 종료의 명령에 대한 결과는 상황판을 통해서 나타내게 된다.

2.2 파라미터 전송 인터페이스

파라미터 전송 인터페이스는 시스템 제어에 필요한 각종 파라미터를 전송하는 인터페이스를 제공한다. 다양한 제어 방법을 지원하기 위해서 사용자와 제어 프로그램은 파라미터를 공유해야 하며 그 정보전송의 확장성이 보장되어야 한다. 본 연구에서 구현된 가상 실험실은 제어 파라미터의 개수가 확장 가능하도록 설계되었으며, 그와 더불어 확장 가능한 파라미터 전송 인터페이스를 제공한다. 자바의 AWT (abstract windowing toolkit)패키지를 바탕으로 구현된 인터페이스는 컨테이너(container) 기반으로 구성되어 있어서 Text Field Component를 쉽게 추가 할 수 있는 장점이 있다.

2.3 시스템 제어기 인터페이스

시스템 제어 인터페이스는 헬리콥터나 도립진자 같은 플랜트 시스템을 제어하는데 필요한 인터페이스를 제공한다. 제안된 가상실험실은 다양한 제어 시스템을 다양한 방법으로 제어하는 도구를 제공하는 것이 목적이다. 따라서, 제어 시스템을 위한 독립된 인터페이스가 필요하다. 기본적으로 주어진 제어 옵션은 플랜트의 제어 시작과 제어 종료 명령이다.

2.4 제어 결과 출력 인터페이스

결과 출력 인터페이스는 가상 실험실의 실험결과를 보여 주는 인터페이스다. 플랜트 시스템과 제어 방법에 따라서 실험 결과로 보여지는 파라미터의 수가 달라지기 때문에, 확장성을 가지는 독립적인 인터페이스가 필요하다. 이 인터페이스는 기본적으로 측정값 그래프 출력 패널(panel) 과 변수값 출력 패널로 구성된다.

3. 시스템 구조

제안된 가상실험실의 시스템 구조는 그림 1과 같다. 서버 컴퓨터는 인터넷 기반 가상실험실을 구축하기 위한 웹서버로 운영된다. 클라이언트 컴퓨터는 제어 시스템과 연결되어 있으며, 인터넷을 통해 서버 컴퓨터와 연결된다. 제어 시스템은 제어 플랜트, 클라이언트 컴퓨터, 제어 플랜트를 중계하는 인터페이스 회로, 그리고 전원 공급기로 구성되며, 제어 시스템의 종류 및 제어 방법에 따라서 조이스틱등의 장비가 추가될 수 있다.

3.1 서버와 클라이언트의 연결 구조

기개발된 가상실험실은 많은 경우 플랜트 시스템을 제어

하는 컴퓨터와 웹서비스를 제공하는 컴퓨터를 분리하여 사용하지 않았다. 그러나, 웹 서버와 제어 프로그램이 혼합된 형태를 갖는 가상실험실은 구현이 비교적 용이하다. 반면, 추가적인 플랜트 시스템이 컴퓨터와 직접 연결이 되어야 하기 때문에 가상 실험실을 통해서 제공 될 수 있는 제어 가능한 플랜트 시스템의 수가 플랜트와 서버 컴퓨터와의 거리와 서버 컴퓨터의 확장성에 제한을 받는다. 본 연구에서 제안된 가상 실험실은 서버 컴퓨터와 클라이언트 컴퓨터를 분리하여 구성하고 이를 인터넷으로 연결함으로써, 제어 시스템의 설치장소에 관계없이 제어 플랜트의 원격제어가 가능하며 사용자의 공간적 위치와 관계없이 가상실험실을 이용할 수 있다. 서버 컴퓨터는 클라이언트 컴퓨터의 목록을 가지고 있으며, 웹을 통해서 사용자에게 목록을 제공한다.

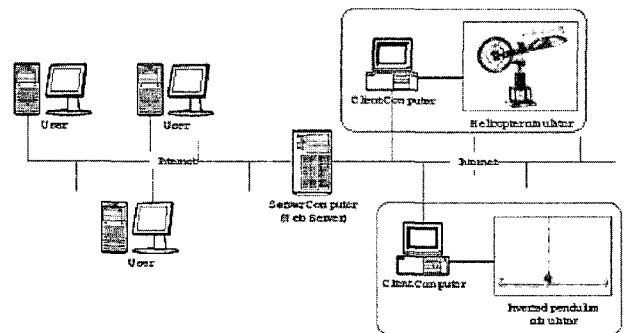


그림 1. 가상실험실의 구성도

Fig. 1 Structure of the VL.

3.2 클라이언트의 내부 구조

클라이언트 컴퓨터와 제어 시스템은 제어 시스템 개발사에서 제공한 통신 인터페이스 카드로 연결되어 있다. 본 연구에서는 2자유도 헬리콥터 시스템과 도립진자 시스템을 가상 실험실에서 제공하는 실험 장치로 사용하였다. 대부분의 실험장치들과 마찬가지로 2자유도 헬리콥터 시스템과 도립진자 시스템을 제작한 Quanser(사)는 PCI(peripheral component interconnect) 인터페이스 보드와 터미널 (terminal) 보드를 제공하여 컴퓨터를 이용한 제어 실험장치를 구현한다. 그림 2는 Quanser(사)의 플랜트를 이용한 가상 실험실의 클라이언트 컴퓨터의 구성도이다. 터미널 보드는 제어 플랜트에 장착된 측정 센서의 연속시간 측정값을 이산화하거나, PCI 인터페이스 보드에서 출력된 디지털 제어 입력을 아날로그 신호로 변환시켜 전원 공급기에 전달하는 역할을 수행한다. PCI 보드는 변환된 디지털 신호를 컴퓨터 시스템이 처리 가능한 데이터 형태로 변형함으로써 가상실험실의 사용자들이 컴퓨터를 활용하여 다양한 제어이론을 검증하기 용이하도록 한다.

4. 응용프로그램 구조

가상실험실 운영 프로그램은 크게 클라이언트 컴퓨터에서 실행되는 플랜트 제어 응용프로그램과 원격제어 응용프로그램, 두 개의 응용프로그램으로 구성된다. 플랜트 제어 프로그램은 클라이언트 컴퓨터와 연결된 제어 시스템을 가상실험실 사용자의 요청에 따라 제어동작을 수행한다. 플랜트 제어 프로그램은 다중 사용자의 요청에 대응해야 하므로 서버-클라이언트 구조상 서버 프로그램의 성격을 가진다.

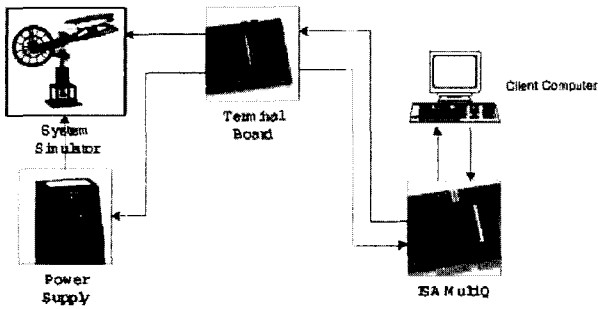


그림 2. 클라이언트 구성도.
Fig. 2 Structure of the client in the VL.

원격제어 프로그램은 가상실험실에서 제공하는 웹페이지에 덧붙여 실행되며, 2장에서 언급한 제어 인터페이스를 사용자에게 제공한다. 제공된 제어 인터페이스를 통해서 입력된 제어 명령은 인터넷 기반 통신을 통해 플랜트 제어 프로그램으로 전달된다.

원격제어 프로그램은 오직 하나의 플랜트 제어 프로그램과 통신을 한다. 반면 플랜트 제어 프로그램은 여러 개의 원격제어 프로그램과 통신한다. 또한 여러 사용자가 웹을 통해 가상실험실에 접속 가능하므로, 원격제어 프로그램은 네트워크 프로그램 구조상 클라이언트 프로그램의 성격을 갖는다. 그림 3은 본 연구에서 개발된 가상실험실의 응용프로그램 구조이다. 원격제어 프로그램은 HTML의 홈페이지 문서와 자바 애플릿으로 작성되며 플랜트 제어 프로그램은 제어 시스템과 연결된다. 플랜트 제어 프로그램은 제어 시스템마다 하나씩 존재하며, 여러 개의 자바 애플릿과 통신한다. 반면 각각의 제어 방법을 제공하는 자바 애플릿은 단지 하나의 서버와 통신한다. 플랜트 제어 프로그램이 일대 다 통신을 수행함으로써 발생하는 통신 충돌 문제는 TCP를 사용함으로써 방지한다.

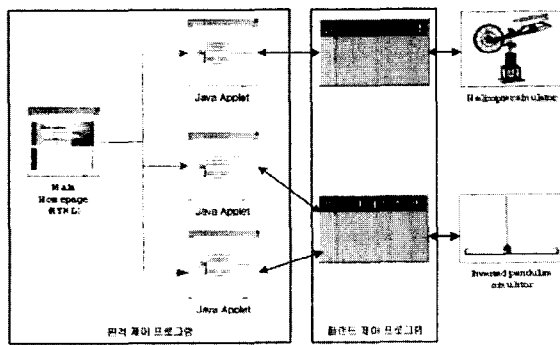


그림 3. 가상실험실 응용프로그램 구성도.
Fig. 3 Structure of the application program in VL.

4.1 프로그램 구성에 따른 언어 선택

가상실험실의 성능과 기능의 구현은 적성 프로그램의 선택에 따라 다르다. 플랜트 제어 프로그램은 여러 하드웨어의 접근기능이 높아야 하며, 원격제어 프로그램은 부가적인 프로그램을 설치하지 않고 인터넷의 개방성을 유지하며 웹 상에서 쉽게 실행되어야 한다.

따라서, 본 논문에서는 위의 조건을 만족시키는 동시에 원격제어 성능을 최대화하기 위해 서로 다른 두 가지 언어를 사용하여 두 가지의 필수 응용프로그램을 구현하였다. 플랜

트 제어 프로그램의 경우, 하드웨어 인식이 가장 좋으며 연산 처리 속도가 빠른 C++을 이용하여 구현하였다. 더욱이 많은 제어실험 장비 개발사에서 제공되는 드라이버들이 대부분 C++로 작성되므로 C++을 이용하면 쉽게 드라이버를 이용하여 하드웨어를 제어할 수 있다. 또한 실시간 하드웨어 제어를 위한 다양한 함수를 제공하고 있어서 실시간 제어를 구현이 용이하다. 두 번째, 원격제어 프로그램은 웹을 통한 실행이 가능해야 하기 때문에 자바언어를 사용하였다. 자바는 컴퓨터의 플랫폼에 관계없이 실행되며, 또한 별다른 설치가 필요 없기 때문에 사용자들의 접근성을 극대화 할 수 있다. 또한 네트워크 통신을 위한 많은 클래스가 지원되며, 확장성을 고려한 인터페이스 구축에 용이한 다양한 클래스들이 지원이 되는 잇점들이 있다. 다만 보안 문제로 인해 네트워크 통신을 이용하는 애플릿의 구현에 약간의 어려움이 있으며 이러한 문제점은 인증 애플릿을 사용함으로써 극복할 수 있다.

4.2 Transmission Control Protocol

플랜트 제어 프로그램과 원격제어 프로그램간의 통신은 인터넷을 기반한 TCP를 이용한다. TCP는 OSI(Open Systems Interconnection) 7층의 전송계층에 해당하는 프로토콜이다. TCP는 데이터의 신뢰성 있는 전송을 보장하며, 여러 제어, 혼잡 제어 등의 고속 데이터 전송을 위한 여러 가지 기술을 제공한다. TCP 프로토콜은 서버와 클라이언트를 가지며, 클라이언트가 서버로 연결을 요청하고 서버가 그 요청을 받아들이면 통신 연결이 성립된다. 이 연결은 통신이 끝날 때까지 유지되며, 서버는 통신 중에 다른 클라이언트의 연결 요청을 수용하지 않는다. 많은 프로그래밍 언어에서 소켓형태로 TCP기반 통신기능을 제공한다. 가상실험실의 원격제어 프로그램과 플랜트 제어 프로그램이 서버-클라이언트 형태이므로 TCP를 이용한 통신 모듈을 가상실험실에 성공적으로 적용할 수 있다. 더구나 신뢰성 있는 통신을 보장하기 때문에 제어 명령에 민감한 플랜트를 제어하는데 적합하다. 또한 서버와 클라이언트의 독립적인 연결을 유지 시켜주기 때문에 여러 클라이언트의 동시 접속을 방지 할 수 있다. TCP 이외에도 UDP(User Datagram Protocol) 프로토콜이 존재하지만, 신뢰성 있는 연결을 보장하지 않기 때문에 가상실험실의 통신 방법으로는 적합하지 않다. 구현된 가상 실험실에서 플랜트 제어 프로그램은 C++ 에서 제공하는 CAsyncSocket 클래스를 이용, 소켓을 생성하였고, 원격제어 프로그램은 자바에서 제공하는 Socket 클래스를 이용하여 소켓을 생성한다.

4.3 모듈화

제어 시스템에 따라서 플랜트 제어 프로그램이 달라지지만, 공통적인 부분이 존재한다. 이러한 부분을 확장 가능하도록 모듈화하면, 프로그램 개발에 편의를 도모할 수 있을 뿐 아니라, 검증된 모듈이 사용가능 하므로 프로그램의 안정성을 높일 수 있다. 구현된 가상 실험실에서는 플랜트 제어 프로그램은 통신 부분과 실시간 제어 부분을 원격제어 프로그램에서는 통신 부분과 데이터 디스플레이 부분을 모듈화 하였다.

4.3.1 플랜트 제어 프로그램

통신 모듈: 플랜트 제어 프로그램의 통신 모듈은 C++의 CAsyncSocket를 상속받는 CClientSock 클래스로 구성된다.

CClientSock의 헤더 파일은 소스1과 같다. CClientSock은 클라이언트가 접속했을 때와 새로운 메시지가 도착했을 때, 이벤트를 발생한다. 제공된 통신 모듈은 새로운 클라이언트가 접속을 했을 때 통신 상태를 점검하고, 제어 상태를 초기화한다. 클라이언트로부터 전송되는 데이터는 통신 명령어에 따라서 해석되며, 플랜트 제어 프로그램은 해석된 명령어에 해당하는 작업을 수행한다. 명령어 해석 부분 또한 통신 모듈로 제공되며, 데이터가 도착했을 때 발생하는 이벤트 함수에 포함된다. 또한 통신 모듈은 클라이언트 프로그램으로 데이터 전송기능을 제공한다.

```
#define WM_RECEIVE_DATA WM_USER+2
class CClientSock : public CAsyncSocket{
public:
    CClientSock();
    void SetWnd(HWND hwnd);
    virtual void OnReceive( int nErrorCode );
public:
    HWND m_pHwnd;};
```

소스 1 CClientSock의 헤더파일.
Source 1 Header file of CClientSock.

실시간 제어 모듈: 대부분의 제어 시스템은 아날로그 시스템이기 때문에 실시간으로 디지털 제어를 수행하기 위해서는 일정한 주기를 갖는 샘플링 모듈이 필요하다. 이 모듈은 전체 프로그램과 관계없이 병렬적으로 수행되어야 한다. C++에서 제공되는 timeSetEvent 함수를 이용하면 이러한 실시간 제어 모듈이 구현 가능하다. timeSetEvent는 임의의 주기를 갖는 이벤트를 발생시키며, 별도의 쓰레드(thread) 상에서 수행되기 때문에 실시간 제어 모듈을 구현하기에 적합한 함수이다. 또한 쓰레드 외부에서 이벤트를 제어할 수 있기 때문에 실시간 제어 상태를 제어 할 수 있다. 소스 2는 timeSetEvent의 선언부이다.

```
nTimerID = timeSetEvent(Delay,
    Resolution, MyTimeProc,
    (DWORD)(CEx3101View *) this,
    TIME_PERIODIC);
```

소스 2 timeSetEvent의 선언부.
Source 2 Declaration of timesetEvent.

4.3.2 원격제어 프로그램

통신 모듈: 원격제어 프로그램의 통신 모듈은 Socket 클래스를 상속받는 mSocket 클래스를 기반한다. 애플릿이 시작되면, mSocket이 초기화되며, 입출력 스트림(stream)이 설정된다. 자바는 새로운 메시지에 대한 이벤트 함수가 별도로 없기 때문에, 서버로부터 전송되는 메시지를 입력받기 위해 입력 버퍼를 모니터링 하는 쓰레드가 요구된다. 새로운 데이터가 입력 버퍼에 들어오면 이들은 내부 명령어에 바탕으로 해석되고 수행된다. 소스 3은 소켓을 생성하고 입출력 스트림을 할당하는 부분을 나타낸다.

```
mSocket=new Socket("165.132.130.97",2000);
try{
    in = new DataInputStream
        ( new BufferedInputStream
        (mSocket.getInputStream()));
    out = new OutputStreamWriter
        (new BufferedOutputStream
        (mSocket.getOutputStream()));
    }
    catch (UnknownHostException t)
    {statusText.setText("open failed : host error");}
```

소스 3 소켓생성 및 입출력 설정.
Source 3 Socket creation.

데이터 출력 모듈: 데이터 출력 모듈은 결과 출력 인터페이스를 구현하기 위한 Text Field 패널과 그래프패널을 제공한다. Text Field 패널은 데이터를 통신 모듈에서 사용된 스레드를 이용하여 실시간 갱신한다. 그래프 패널은 재 정의된 paint 함수를 이용한다.

4.4 내부 명령어

가상실험실에서 사용되는 다중 명령어 통신을 구현하기 위해서는 이에 맞는 내부 명령어가 필요하다. 내부 명령어는 문장이나 특수 문자 등을 이용하여 정의되며, 제어 시스템에 따라서 달라지는 특수 명령어와 제어 시스템과 관계없이 일정한 기본 명령어로 나눌 수 있다. 2자유도 헬리콥터 시스템의 특수 명령어는 표 1과 같다. 특수 명령어는 제어 명령과 연결 명령에 관련된 명령어들이며, 기본 명령어는 모니터링 및 실시간 데이터 전송을 위한 명령어들이다. 이들은 실시간으로 데이터를 전송하고 해석하는데 사용되므로 일반 명령어들은 대부분 짧은 문자로 구현된다. 특수 명령어는 구분을 위하여 명령과 연관된 긴 단어를 사용한다.

표 1 헬리콥터 제어시스템 내부 명령어.
Table 1 Internal command of the helicopter

특수 명령어	설명
Startopenloop	System monitoring start
PitchUp	Pitch volt increase by 0.1V
PitchDown	Pitch volt decrease by 0.1V
YawUp	Yaw volt increase by 0.1V
YawDown	Yaw volt decrease 0.1V
STOPhelicopter	Stop operating helicopter
Setparam	Set parameters of controller
FuzzyControlStart	Start fuzzy control and monitoring
FuzzyControlStop	Stop fuzzy control and monitoring

4.5 보안

원격제어 프로그램을 구현하기 위해 사용된 자바언어의 애플릿은 자체의 높은 접근성으로 인하여 다음과 같은 보안 제한을 받는다.

- 애플릿은 로컬 컴퓨터의 디스크를 읽거나 쓸 수 없다.
- 애플릿은 원래 자신이 위치한 컴퓨터 외의 다른 컴퓨터와

네트워크 연결이 불가능하다.

- 애플릿은 로컬 컴퓨터에서 프로그램을 실행이 불가능하다.
- 애플릿은 로컬 컴퓨터에서 공유 라이브러리를 호출할 수 없다.

본 연구에서 개발된 가상실험실은 애플릿 프로그램을 이용한 소켓통신을 수행하므로, 위의 두 번째 보안 정책에 의해서 제한을 받아 그 구현이 난해하다. 그러나, 본 연구에서는 인증 애플릿을 이용하여 이 문제를 해결하였다. 인증 애플릿은 정책 파일과 서명을 함께 애플릿과 제시함으로써 보안 인증을 받는 방법이다. 정책 파일(policy file)은 정책 파일과 함께 실행되는 모든 애플릿이나 애플리케이션에 서명이 필요하도록 정의되어질 수 있다. 서명은 애플릿이나 애플리케이션이 신뢰성 있는 소스이고 정책 파일에서 허가된 권한을 가지고 실행되는 것을 신뢰할 수 있다고 확인하는 방법이다. 인증 애플릿은 Microsoft SDK for JAVA 프로그램을 이용하여 생성한다. 순서대로 명령어를 수행 시켜야 하며, Microsoft SDK for JAVA가 설치되어 있어야 한다. 마지막으로 인증된 애플릿이 실행되기 위해서는 최종 사용자의 허가가 있어야 한다. 그림 4는 사용자의 허가를 받기 위한 보안 경고 메시지이다. 신뢰성 있는 인증을 받기 위해서는 국제 인증기관에 인증 허가를 신청해야 한다.

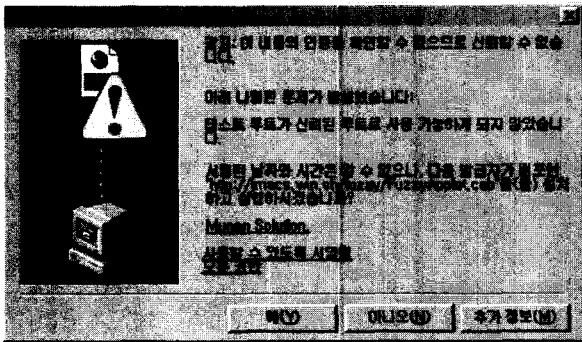


그림 4. 사용자 인증 보안 메시지.
Fig. 4 Message of user certification.

5. 가상실험실 구축

본 연구는 2자유도 헬리콥터 시스템과 도립진자 시스템을 가상실험실에서 활용 가능한 실험장치로 도입하였다. 2자유도 헬리콥터 시스템의 경우 개루프 제어기법과 퍼지 모델 기반 분산 제어기법을 기본적으로 제공한다. 가상실험실의 사용자는 새롭게 개발된 제어이득을 입력함으로써 자신이 개발된 제어 이론을 실험을 통하여 평가할 수 있다. 도립진자 시스템은 상태회환 제어기법을 지원한다. 서버는 일반적인 웹 서버를 이용할 수 있다. 그림 5는 가상실험실 홈페이지의 시작 화면이며 Introduction, Virtual LAB, BBS 세 가지의 메뉴를 제공한다. 두 개의 실험 장치와 각각의 제어프로그램이 실행되는 컴퓨터는 실험장치 제작사인 Quanser(사)에서 제공하는 Multi-Q 보드로 연결된다.

2자유도 헬리콥터 시스템: 헬리콥터 시스템은 직류모터에 의해 구동되는 두 개의 프로펠러와 두 개의 센서를 장착한다. 가상실험실은 기본적으로 개루프 제어 및 퍼지 분산 제

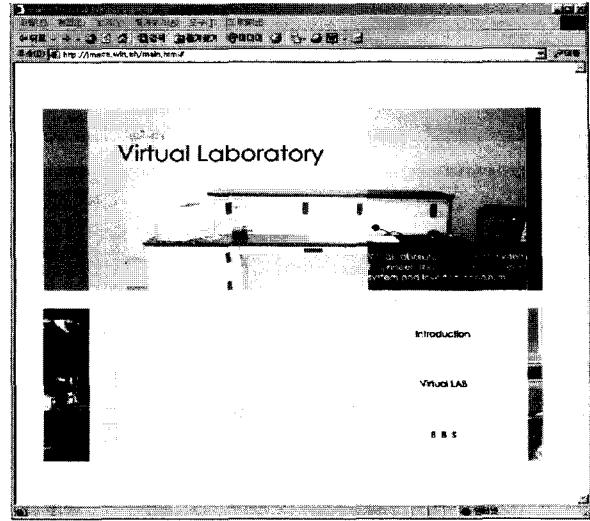


그림 5. 가상실험실 시작화면.
Fig. 5 Initial page of the VL.

어 방식을 두 가지 애플릿으로 제공하여, 웹을 통하여 제어 방법을 선택 할 수 있다. 그림 6은 가상실험실 홈페이지의 헬리콥터 페이지이다. 가상실험실 사용자는 헬리콥터에 장착된 2개의 모터의 전압을 조절함으로써 헬리콥터의 자세를 제어하며, 그 결과를 그래프에 실시간으로 확인할 수 있다. 그림 7은 통해 제공된 헬리콥터 개루프 제어기의 인터페이스와 실험 결과를 보여주고 있다.

구현된 웹 인터페이스는 사용자로 하여금 제어 파라미터 값을 입력 할 수 있도록 구성되어 있다. 단, 초기 헬리콥터의 상하 각이 0인 점을 감안하여, -15도의 오프셋을 고려하였다. 그림 7은 구현된 퍼지 분산 제어기 시스템의 실험 장면이다.

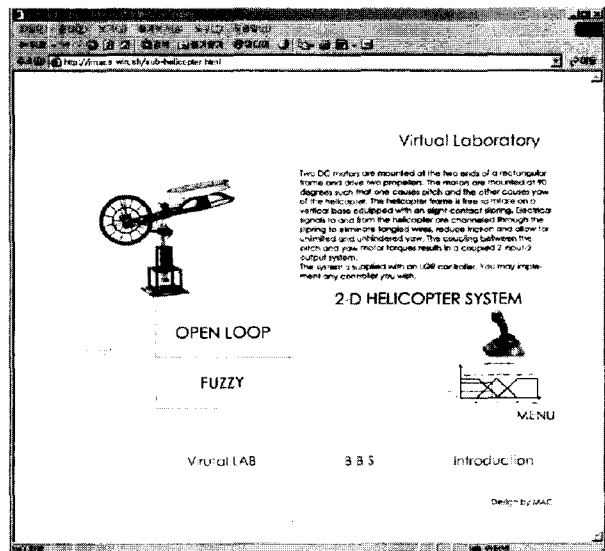


그림 6. 가상실험실의 2자유도 헬리콥터 시스템의 실험 페이지.

Fig. 6 Helicopter system with 2 degree of freedom in VL

도립진자 시스템: 도립진자 시스템은 막대와 카트(cart)로 구성된다. 도립진자 시스템은 한 개의 모터와 카트의 위치와 막대의 기울기를 측정하는 두 개의 센서로 구성된다. 측정된 값을 multi-Q 보드를 통해 컴퓨터로 전송한다. 한 개의 모터는 카트의 위치를 제어한다. 그림 8는 웹을 통해 제공되는 가상실험실의 도립진자 페이지이다. 구현된 도립진자의 제어 인터페이스는 제어 파라미터를 입력받도록 설계되어 있다. 입력된 제어 파라미터는 웹을 통해서 제어 플랫폼 프로그램으로 전달되며, 사용자는 전달된 제어 파라미터를 이용하여 도립진자 시스템을 제어할 수 있다. 그림 9는 구현된 웹 인터페이스를 이용하여 도립진자 시스템을 제어하는 모습이다.

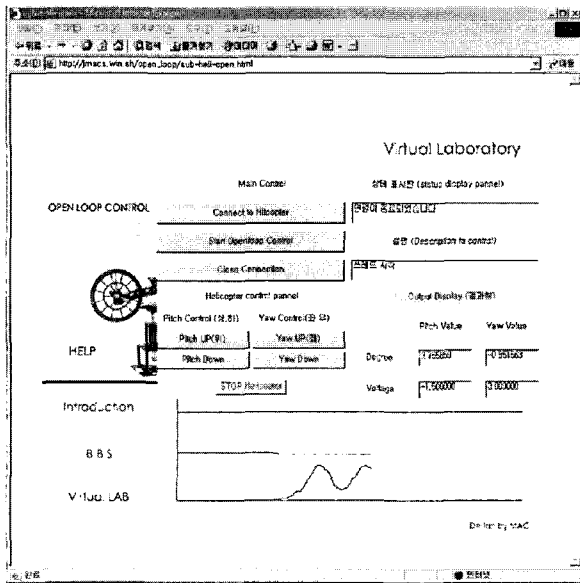


그림 7. 2자유도 헬리콥터 시스템의 개루프 실험 페이지
Fig. 7 Open-loop control page of the 2 degree of freedom helicopter system in VL.

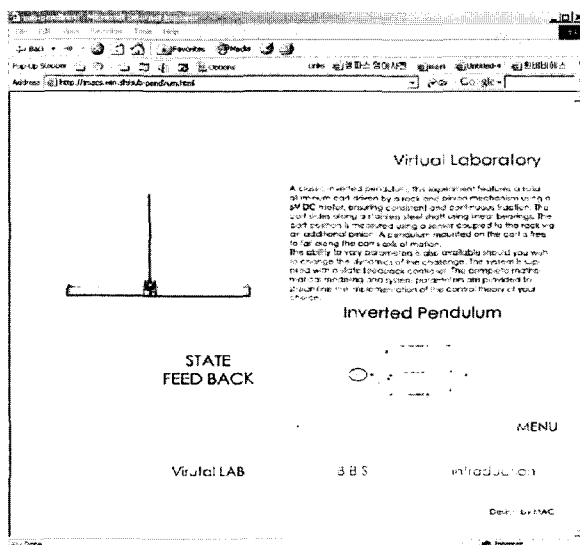


그림 8. 가상실험실의 도립진자 시스템의 실험 페이지.
Fig. 8 State feedback page of the inverted pendulum system in VL.

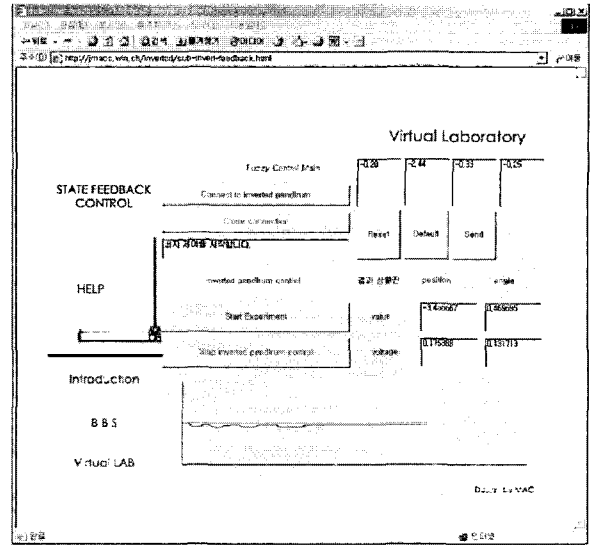


그림 9. 도립진자 시스템의 상태 피드백 제어 실험 페이지.
Fig. 9 State feedback page of the inverted pendulum system in VL.

6. 결론

본 논문은 가상실험실의 구현을 논의하였다. 실험장치의 확장성을 고려하여 전체적인 시스템은 서버-클라이언트의 형태로 구성되며 이들은 TCP를 이용하여 건설한 통신을 수행한다. 플랫폼 제어 프로그램은 시스템의 이식성을 고려하여 C++을 사용하여 작성되었으며, 원격제어 프로그램은 사용자의 접근성을 극대화하기 위하여 컴퓨터의 플랫폼에 관계없이 실행되는 자바를 사용하였다. 제어 인터페이스와 규격화된 통신 모듈을 제공함으로써 쉽게 다양한 제어 방법을 구현할 수 있다. 이와 더불어 가상실험실의 확장성을 유지하기 위하여 응용프로그램을 모듈화 하였으며 손쉽게 다양한 실험 장비를 가상실험실에 추가할 수 있다. 따라서, 본 논문에서 제안된 가상실험실을 활용하면 각 대학에서 보유하고 있는 각종 실험장비를 가상실험실로 통합함으로써 연구비, 연구시간의 절약 등의 기대효과가 예상된다.

참고 문헌

- [1] A. C. Weaver, "Monitoring and Control Using the Internet and Java", Industrial Electronics Society, IECON, Vol. 3, pp. 1152-1158, 1999.
- [2] R. B. Sepe, "Web-based Virtual Engineering Laboratory for Real-Time Control of a Hybrid Electric Vehicle Starter/Alternator," Digital Avionics Systems Conference, Vol. 2, pp. 3-1-3-7, 1999.
- [3] C. Rohrig and A. Jochheim, "The Virtual Lab for Controlling Real Experiments via Internet," Proc. of the 1999 IEEE Int. Symp. on Computer Aided Control System Design, pp. 279 -284, 1999.
- [4] Y. F. Ho, H. Masuda, and H. Oda, L. W. Stark, "Distributed Control for Teleoperations," Proc.

- 1999 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics, pp. 323-328, 1999.
- [5] A. Leleve, P. Fraisse, P. Dauchez, and P.F. Pierrot, "Modeling and Simulation of Robotic Tasks Teleoperated through the Internet," IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics, pp. 299-304, 1999.
- [6] R. B. Sepe, M. Chamberland, and N. Short, "Web-based Virtual Engineering Laboratory for a Hybrid Electric Vehicle Starter/Alternator," Industry Applications Conference, 1999. Thirty-Fourth IAS Annual Meeting. Vol. 4, pp. 2642-2648, 1999.
- [7] G. Bruzzone, R. Bono, M. Caccia, and G. Veruggio, "Internet Mission Control of the Romeo Unmanned Underwater Vehicle Using the Coral Mission Controller," OCEANS '99 MTS/IEEE. Riding the Crest into the 21st Century, Vol. 3, pp. 1081-1087, 1999.
- [8] J. W. Overstreet and A. Tzes, "Internet-based Client/Server Virtual Instrument Designs for Real-time Remote-access Control Engineering Laboratory," American Control Conference, Vol. 2, pp. 1472-1476, 1999.
- [9] Y. Piguet and D. Gillet, "Java-based Remote Experimentation for Control Algorithms Prototyping," American Control Conference, Vol. 2, pp. 1465-1469, 1999.
- [10] C. P. Young, W. L. Juang, and M. J. Devaney, "Real-time Intranet Controlled Virtual Instrument Multiple-circuit Power Monitoring," Instrumentation and Measurement Technology Conference, Vol. 2, pp. 673 -677, 1999.
- [11] K. B. Lee and R. D. Schneeman, "Internet-based Distributed Measurement and Control Applications," IEEE Instrumentation and Measurement Magazine, Vol. 22, pp. 23-27, 1999.
- [12] J. W. Overstreet and A. Tzes, "An Internet-based Real-time Control Engineering Laboratory," IEEE Control Systems Magazine, Vol.19, pp. 19-34, 1999.
- [13] M. Bertocco, F. Ferraris, C. Offelli, and Parvis, "A Client-server Architecture for Distributed Measurement Systems," IEEE Trans. on Instrumentation and Measurement Vol. 47, pp. 1143-1148, 1998.

저 자 소 개

주영훈(Young Hoon Joo)

제 13권 3호(2003년 6월호) 참조



김문환(Moon Hwan Kim)

2000년~현재 : 연세대학교 전기전자공학부 학사과정.

관심분야 : TS 퍼지 시스템, 퍼지 PID 제어, 유전 알고리즘

Phone : 02-2123-2773

Fax : 02-362-4539

E-mail : jmacs@control.yonsei.ac.kr

이호재(Ho Jae Lee)

제 13권 3호(2003년 6월호) 참조

박진배(Jin Bae Park)

제 13권 3호(2003년 6월호) 참조