

# 웹기반 범용 다물체 동역학 시뮬레이션 시스템 개발

한형석<sup>#</sup>, 이재경<sup>\*</sup>

## Development of a Web-based Dynamic Simulation System for Multibody Systems

Hyung Suk Han<sup>#</sup>, Jai Kyung Lee<sup>\*</sup>

### ABSTRACT

A Web-based dynamic simulation system, called O-DYN, for multibody dynamic systems is developed. All the interfaces of the system are accessible via Web browsers, such as Netscape or Explorer. The system uses a block-diagram type O-DYN/Modeler developed in JAVA Applet as a preprocessor. The O-DYN postprocessor composed of O-DYN/Plotter and O-DYN/Animator is developed in JAVA Applet. The O-DYN/Solver for predicting the dynamic behavior is run on the server. Anyone who wants to simulate the dynamics of multibody systems or share results data can access the analysis system over the Internet regardless of their OS, platform, or location.

**Key Words** : Multibody system(다물체 계), Web-based(웹기반), JAVA(자바), Internet(인터넷), Web Browser(웹 브라우저)

### 1. 서론

다물체 계의 동적 시뮬레이션은 다물체 동역학 시뮬레이션 기술의 진보와 상용 소프트웨어의 발전으로 일반화되고 있다. 다물체 동역학은 현재 자동차, 중공업, 항공, 전자, 생명공학 등 다양한 분야에서 활발히 적용되고 있다. 최근에는 가상공학에 있어서 다물체 동역학 시뮬레이션 기술은 중요한 한 부분을 차지하고 있으며 그 중요성이 점차 증가하고 있다. 그러므로 다물체 동역학 시뮬레이션은 이제 일부 전문가만의 영역이 아니라고 할 수 있다. 대상 기계의 개발에 관련된 모든 기술자가 쉽게 다물체 동역학 시뮬레이션 과정에 참

여하고 시뮬레이션 결과를 공유하여 협업을 이루는 것이 바람직할 것이다.

현재 인터넷과 웹 응용기술의 발달은 정보의 공유와 유통에 새로운 기회를 제공하고 있다. 웹 응용은 사용자의 하드웨어, 소프트웨어의 환경이나 장소의 한계를 극복할 수 있는 장점을 갖는다. 더 나아가 웹을 통한 협업을 달성할 수 있다. 웹 응용 기술은 시뮬레이션 분야에서도 새로운 기회를 제공하고 있다. 산업공학 분야에서의 웹기반 시뮬레이션 활동은 활발히 이루어지고 있으나 기계공학 분야에 있어서는 아직 시작 단계라 할 수 있다.<sup>1-7</sup> 웹 환경에서의 기계 공학적 시뮬레이션도 수년 내 일반화 될 것으로 사료된다. 웹기반 시스

<sup>#</sup> 접수일: 2003 년 2 월 21 일; 게재승인일: 2003 년 7 월 11 일  
<sup>#</sup> 교신저자: 한국기계연구원  
Email:hshan@kimm.re.kr, Tel: (042) 868-7814  
<sup>\*</sup> 한국기계연구원

템은 기계공학자의 사용자 컴퓨터의 환경이나 사용자 위치에 제한이 없이 원하는 공학 계산을 웹 서버와 웹 브라우저를 통하여 수행이 가능하게 된다. 더 나아가 웹서버간 통신에 의한 분산 시뮬레이션으로의 확장이 가능하다. 다수의 사용자가 동시에 결과를 공유하며 토의할 수 있어 협업 및 동시공학이 가능하게 된다. 또한, 시뮬레이션 결과 및 관련 데이터를 서버에 DB 화하여 자료의 재사용성도 증가할 것이다.

현재 웹기반 다물체 동역학 시뮬레이션 시스템은 발표된 사례가 없다. 상용화된 다물체 동역학 시뮬레이션 시스템으로는 DADS<sup>8</sup>, ADAMS<sup>9</sup>, RecurDyn<sup>10</sup>, SIMPACK<sup>11</sup> 등이 있으며 모두 독립형(Standalone) 방식이다. 이러한 것들은 시뮬레이션 시스템이 설치된 컴퓨터에서만 사용이 가능하거나 보조 프로그램을 이용하여 원격으로 사용할 수 있다. 독립형은 설치된 컴퓨터에서만 사용할 수 있기 때문에 다수의 사용자가 이용하기 어렵다. 최근에는 관련 기술자가 지리적으로 광범위하게 분포하는 경우가 많기 때문에 독립형은 시뮬레이션에 이용된 데이터나 결과를 공유하기 어렵다. 독립형은 또한 데이터가 독립 컴퓨터에 저장되기 때문에 기술 정보의 체계적 관리가 되지 않는다. 상기에서 언급한 상용 시스템들은 또한 1 명의 사용자를 기준으로 하며 사용자가 증가함에 따라 비례적으로 비용이 증가한다. 최근에는 일부 대형 해석용 소프트웨어 제조업체에서 보조 프로그램을 이용한 유료 응용프로그램제공(ASP) 서비스를 시작하는 것으로 알려져 있다. 이러한 서비스의 도입을 볼 때 공학용 시뮬레이션 시스템도 독립형에서 웹기반 공학 시스템으로의 전환이 이루어지는 것을 알 수 있다.

본 논문에서는 웹기반 범용 다물체 동역학 시뮬레이션 시스템을 소개한다. 본 논문은 기계공학자 측면에서 웹 응용 기술을 다물체 동역학 시뮬레이션에 적용하는 방안을 제시하기 위하여 시스템의 전반적 구조와 주요 구현기술을 소개하는데 목적을 둔다. Fig. 1 은 본 논문에서 목표로 하는 시스템을 간략히 보여주고 있다. Fig. 1 에서와 같이 다물체 동역학 해석기는 웹서버에 설치되며 인터넷을 통하여 실행시키게 된다. 모든 공학적 계산 및 데이터 관리는 이 웹서버에서 이루어진다. 사용자와의 모든 인터페이스는 웹 브라우저를 통하여 이루어지게 된다. 사용자는 웹 브라우저 상에서

다물체 계의 모델링, 해석, 결과 분석을 수행한다. 결과적으로 다수의 사용자가 사용자 컴퓨터 환경이나 위치에 상관없이 웹을 통하여 다물체 동역학 시뮬레이션을 수행할 수 있다. 또한 공학적 정보를 다수의 사용자가 위치에 상관없이 동일 또는 비동일 시간에 공유할 수 있어 협업 및 동시공학의 구현이 가능하게 된다.

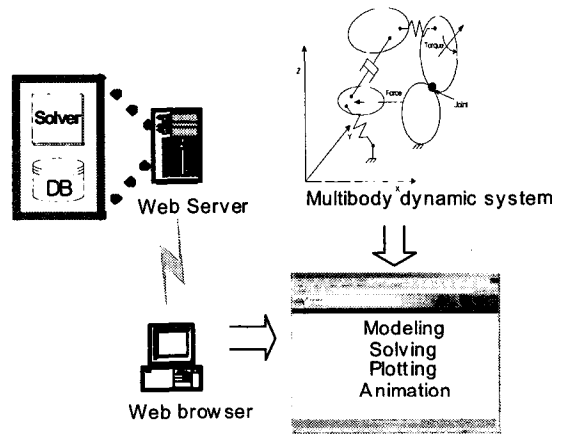


Fig. 1 Target Web-based multibody dynamic simulation system

## 2. 시스템 개발

### 2.1 시스템 구조

Fig. 2 는 본 논문에서 소개하는 시스템 O-DYN 의 구조를 보여주고 있다. 시스템은 크게 PRE Module, POST Module, JOB Control Module, SOLVER Module 로 구성된다. PRE Module 은 사용자가 O-DYN 을 이용하여 시뮬레이션 하려는 대상을 정의하는 기능을 수행하며 동역학 해석기 O-DYN /Solver 의 입력데이터(모델 데이터)를 생성하는 O-DYN/Modeler 로 구성된다. O-DYN/Modeler 는 모델 데이터의 생성/검색/수정/삭제 기능을 제공한다. POST Module 은 O-DYN 의 시뮬레이션 결과에 대한 분석 및 가시화 기능을 제공하며 O-DYN/Plotter 와 O-DYN/Animator 로 구성된다. O-DYN/Plotter 는 시뮬레이션 결과의 분석을 위한 2D/3D 그래프를 제공하며, O-DYN/Animator 는 다물체 계의 거동을 3D 애니메이션으로 제공한다. JOB Control Module 은 PRE Module, POST Module 에서 요청된 작업을 관리하는 기능을 제공한다. PRE Module 에 의한 시

플레이션 요청, 시물레이션 요청에 따른 SOLVER Module 의 구동 및 결과처리, POST Module 에 시물레이션 결과의 전달, 사용자 작업 이력관리(job history)를 담당하는 Job Manager, 사용자 관리 및 인증을 담당하는 User Manager, 사용자의 작업 데이터(시물레이션 모델, 결과데이터, 작업이력 데이터) 관리를 담당하는 Data Manager 로 구성된다. SOLVER Module 은 JOB Control Module 을 통해 전달된 사용자의 모델 데이터를 이용하여 동적 시물레이션을 수행하며 다물체 계의 동역학 해석기 O-DYN/Solver 와 이를 관리하는 Solver Manager 로 구성된다.

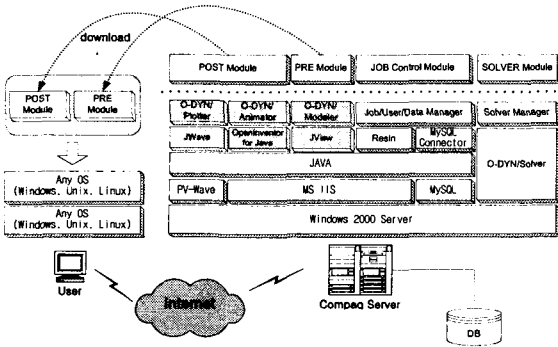


Fig. 2 System architecture

## 2.2 시스템 운영 및 개발환경

Fig. 2 에서와 같이 사용자 인터페이스인 PRE Module, POST Module 은 사용자의 웹브라우저에서 실행되며, JOB Control Module, SOLVER Module 은 서버에서 운영된다. 사용자는 운영체제(OS)나 웹브라우저에 구매 받지 않고 JAVA 를 이용할 수 있는 웹브라우저를 사용하여 시스템에 접속한다. 시스템에 접속한 사용자는 먼저 사용자 인증과정을 거친 후, 동역학 시물레이션을 수행하기 위하여 PRE Module 을 선택하거나 시물레이션 결과의 분석 및 가시화를 수행하기 위하여 POST Module 을 선택한다. 선택된 작업에 따라 Fig. 2 에서 회색으로 표시된 O-DYN/Modeler, O-DYN/Plotter, O-DYN/Animator 가 사용자 웹브라우저로 전송된다. 지원 가능한 JAVA 버전은 JDK 1.3.1<sup>12</sup> 이며, PRE Module 과 POST Module 에서 사용된 OpenInventor for JAVA<sup>13</sup>, JWAVE<sup>14</sup>, JViews<sup>15</sup> 런타임 라이브러리가 사용자 웹브라우저에 전송되며 별도의 설치과정은

없다. 사용자의 작업이 처리되는 서버는 하드웨어로 Compaq ML-570 Server, 운영체제는 Windows 2000 Server 를 사용하였고 O-DYN 을 위하여 운영되는 소프트웨어는 다음과 같다. 데이터베이스 서버는 MySQL, 그래프 서버는 PV-Wave<sup>14</sup>, Servlet/JSP(JAVA Server Page) 엔진은 Resin<sup>16</sup>, 동역학 시물레이션을 위한 해석기는 자체 개발한 O-DYN/Solver, 웹서버는 MS-ISS 가 서버쪽에서 운영된다. 웹서버인 MS-ISS 가 PRE Module, POST Module 을 사용자에게 전송하고 JOB Control Module 을 제공한다. O-DYN 시스템은 웹서버와 운영체제를 Microsoft 사 제품을 사용하였지만 시스템 개발에 JAVA, ANSI C++를 사용하였기 때문에 웹서버가 설치될 수 있는 환경이라면 포팅이 가능한 장점이 있다. O-DYN 은 웹기반 시스템이기 때문에 다수의 사용자가 동시에 시스템을 이용할 수 있기 때문에 동시 사용자 수의 제한 및 로드 밸런싱(load balancing) 문제가 발생할 수 있다. 로드 밸런싱 및 시스템 동시 사용자 수의 제한은 서버의 하드웨어 성능, 네트워크 속도, 사용된 소프트웨어의 사용자 라이선스 수에 제한을 받는다. 동시 사용자 수는 시스템에 연결하여 각 모듈을 동시에 사용하는 사용자 수이며 O-DYN/Modeler, O-DYN/Animator 는 사용자 웹브라우저로 전송된 후 수행되므로 시스템에 영향을 주지 않지만, O-DYN/Plotter 는 사용자 간의 협업시 그래프 서버를 이용하기 때문에 그래프 서버의 라이선스 수와 연관된다. 또한 인터넷으로 연결되어 시스템을 사용하기 때문에 네트워크 속도와 전송량이 중요하며 각 모듈별 크기는 다음과 같다. O-DYN/Modeler 는 GUI 라이브러리 JViews 를 포함한 기본 모듈이 대략 2MB, O-DYN/Plotter 는 그래프 라이브러리 JWAVE 를 포함하여 기본 모듈이 300KB, O-DYN/Animator 는 그래픽 라이브러리 OpenInventor 를 포함하여 기본 모듈이 3MB 정도이다. 각 기본모듈은 사용자 웹브라우저의 캐쉬에 저장되어 매 사용시마다 전송되지 않으므로 각 모듈별 실제 사용되는 데이터 크기가 전송량을 결정한다. O-DYN/Modeler 는 사용자의 시물레이션 모델에 따라 결정되나 모델 데이터의 크기는 매우 적다. O-DYN/Plotter, O-DYN/Animator 는 결과 데이터의 크기가 전송량을 결정하며, 특히 O-DYN/Animator 는 애니메이션을 위한 영상정보를 전송 받으므로 전송량이 많아진다. 시스템의 시험적 운영결과 영상정보의 분할전송 및 압축전

송 방법을 도입하여 전송 속도를 개선시켜야 될 것으로 사료된다. 웹기반 공학 시스템은 인터넷의 특성상 불특정 다수가 시스템에 접속할 수 있으므로 허가된 사용자들만이 시스템을 이용할 수 있는 방법이 제시되어야 한다. O-DYN 에서는 네트워크 보안성(방화벽이나 기타 네트워크 레벨의 보안성)이 확보되었다는 가정하에 사용자에게 id 와 password 를 부여하고 사용자의 네트워크 주소(IP 주소)를 서버에 등록하는 2 단계의 인증을 수행한다. 1 단계에서는 미 등록된 네트워크 주소로부터의 시스템 연결은 거부되며 2 단계에서는 등록된 네트워크 주소라도 사용자 id 와 password 를 검사하여 일치하지 않을 경우엔 거부된다. 또한, 인증을 거쳐 시스템의 사용이 허가되었다더라도 시스템의 모든 사용자가 작업데이터를 공유하는 것은 바람직하지 않으며, O-DYN 에서는 사용자별로 작업데이터를 관리하고 사용자가 자신과 작업데이터를 공유할 사람을 지정하는 방법을 사용한다. 시스템의 시험적 운영결과 작업데이터의 공유방법, 버전닝(versioning), 작업그룹 개념의 도입 등 협업 작업의 개념을 확대시켜야 될 것으로 사료된다. 이러한 운영환경 하에서 O-DYN 이 웹기반 공학 시스템으로의 장점은 다음과 같다. 첫째, 사용자 컴퓨터에 별도의 S/W 를 설치할 필요가 없으므로 S/W 구입비용과 설치비용을 절감할 수 있다. 둘째, S/W 의 업그레이드가 서버측에서만 이루어지므로 업그레이드 비용과 시간을 절감할 수 있다. 셋째, 자체 개발한 해석기를 이용하기 때문에 고가인 해석기 사용권 문제로부터 자유롭다. 넷째, 각 사용자 컴퓨터가 고성능일 필요가 없으며 고성능의 서버만을 유지함으로써 비용절감 및 효율적인 사용이 가능하다. 다섯째, 작업결과가 서버에 저장되고 관리되기 때문에 중요 작업데이터에 대한 관리 및 보안성이 강화되며 작업데이터에 대한 공유가 가능하다. 여섯째, 정보를 동기 또는 비동기적으로 다수의 사용자가 공유할 수 있어 협업 및 동시공학 지원이 가능하다.

Table 1 은 시스템의 개발 및 운영에 이용되는 하드웨어, 소프트웨어를 보여주고 있다. 시스템 개발에 사용된 프로그래밍 언어는 C++, JAVA 이며 C++는 동역학 해석기 O-DYN/Solver 개발에 사용되었으며, O-DYN/Solver 를 제외한 시스템 모듈들은 JAVA 로 개발되었다. 웹 프로그래밍 기법으로는 웹 표준인 HTML 과 JAVA Applet, JSP, Servlet 을

사용하였다. 사용자 인터페이스인 PRE Module, POST Module 은 JAVA Applet 으로 개발되었으며, 해석기 O-DYN/Solver 를 제외한 웹서버쪽의 시스템 모듈은 JAVA Servlet 과 JSP 로 개발되었다. 사용자 인터페이스의 개발에서 사용한 JAVA Applet 의 특징은 다음과 같다. JAVA Applet 은 웹 브라우저에 부착되어 ActiveX 처럼 사용되며, 별도의 프로그램 설치가 필요 없이 필요한 모듈은 다운로드 받아 사용할 수 있는 장점이 있다. 그러나, JAVA Applet 은 보안상 사용자 컴퓨터의 파일 제어 및 자신이 위치한 웹서버 이외의 시스템과의 통신이 불가능하다. 따라서, 시스템의 모든 데이터는 웹서버의 DB, 파일 시스템에서 관리되어야 하며, 이를 통해 데이터의 체계적 관리와 보안성을 확보할 수 있다. 반면, 사용자 컴퓨터의 데이터가 서버로 업로드 되어야 할 경우(사용자 curve data, 형상정보 등), JAVA Applet 의 보안 제약상 업로드가 불가능하며 JSP 방식의 파일 업로더를 사용하였다. 웹서버쪽의 시스템 모듈의 개발에 사용된 JAVA Servlet 의 특징은 다음과 같다. JAVA Servlet 은 JAVA Applet 이 처리할 수 없는 서버의 응용 프로그램 실행이나 파일처리를 수행할 수 있으며, JAVA Applet 과 통신을 통하여 데이터를 주고 받을 수 있으며 세션과 쿠키 제어를 통하여 사용자 인증을 제공할 수 있다. 현재 O-DYN 시스템은 단일 시스템에서 수행되지만, 향후 웹서버 시스템, 데이터베이스 관리 시스템, 동역학 해석 시스템으로 분리되어 확장 운영될 수 있으며, 이를 위해서는 JAVA Applet 방식의 사용자 인터페이스가 앞에서 언급한 보안제약을 극복할 수 있어야 한다. 즉, 데이터베이스 서버, 동역학 해석기가 웹서버와 다른 시스템에 위치해 있으면, 보안제약으로 사용자 인터페이스인 PRE Module, POST Module 이 데이터베이스 서버, 동역학 해석기를 사용할 수 없는데 JAVA Servlet 을 통하여 이를 해결할 수 있다.

프로그래밍 개발툴은 Visual C++ 6.0, JDK 1.3 을 사용하였으며 ANSI C++을 기준으로 개발하였기 때문에 Visual C++대신 일반적인 C++ 컴파일러를 이용할 수 있다. 사용자 정보 및 사용자의 작업데이터(모델 데이터, 결과 데이터, 작업이력 데이터), 다물체 모델링 요소, 로그 데이터를 관리하기 위해 사용된 MySQL 은 관계형(Relational) 데이터베이스 관리 시스템으로 별도의 비용 없이 사용할 수 있고 동시에 다수의 사용자가 데이터를 공유할

수 있으며 데이터의 백업이나 관리가 용이한 장점이 있다. 시스템 개발에 사용된 라이브러리는 다음과 같다. GUI 라이브러리 JViews 는 블록 다이어그램 방식의 사용자 인터페이스를 개발하는데 사용되었으며, 그래프 라이브러리 JWAVE 는 PV-Wave 의 JAVA 버전으로 수치계산 루틴인 IMSL 을 사용할 수 있고 사용자 웹브라우저에 표현 가능한 결과 데이터만 전송하므로 네트워크 부하 및 효율성을 높여 줄 수 있으며, 협업을 위한 기본 기능을 제공하는 장점이 있다. 3D 그래픽스 라이브러리 OpenInventor for Java 는 3D 표준인 OpenGL 에 기반한 라이브러리로 OpenGL 이나 JAVA3D 를 사용하였을때 보다 다양한 기능과 효율적인 실행속도를 보장하며 다수의 CAD 프로그램에서 제공하는 VRML(Virtual Reality Modelling Language) 형식과 iv(OpenInventor File Format) 형식을 형상정보로 읽어 들일 수 있는 장점이 있다.

Table 1 System development environment

OS	Windows 2000 Server
H/W	Compaq Server, 1 CPU
Web Server	MS IIS 5.0
Servlet/JSP Engine	Resin 2.1
Programming	Visual C++ 6.0 with SP4
Tools	JDK 1.3
DBMS	MySQL 3.23
Library	JWAVE 3.5 (PV-Wave 7.5 포함) OpenInventor for Java 3.6 JViews 5.5

### 2.3 PRE Module 개발

PRE Module 은 웹브라우저 상에서 블록다이어그램 형태의 사용자 인터페이스인 O-DYN/Modeler 를 제공한다. O-DYN/Modeler 는 GUI 라이브러리 JView 를 이용하여 JAVA Applet 방식으로 개발되었으며, 사용자간의 모델 데이터 공유가 가능하다. 이때 동시에 같은 모델 데이터를 수정하는 것을 방지하기 위해 모델 데이터의 생성자만이 모델 데이터를 수정할 수 있으며, 공유가 허가된 타 사용자는 읽기만 허용된다. Fig. 3 은 O-DYN/Modeler 의 주요 기능과 작업흐름 그리고 이와 관련된 데이터 베이스를 나타낸다. O-DYN/Modeler 의 운영 시나리오는 다음과 같다. 사용자 인증을 마친 사용자는 O-DYN/Modeler 를 전송 받는다. O-DYN/Modeler

는 모델링에 필요한 다물체 요소 정보를 DB 에서 읽어 동적으로 화면 왼쪽의 다물체 요소 트리를 구성한다. Fig. 4 는 O-DYN/Modeler 화면 예이다. 사용자는 화면 왼쪽의 요소 트리에서 필요한 요소를 선택하여 다물체 체계를 모델링하고 모델 데이터를 DB 에 저장한다. 각 요소 파라메타 입력은 블록을 클릭한 후 Fig. 5 의 좌측 하단과 같이 입력한다. 모델링 검토를 마친 사용자가 동적 해석을 위해서 'Analysis'를 선택하면 Job Manager 가 수행되어 동역학 시뮬레이션을 수행할 수 있다. 개발된 O-DYN/Modeler 는 모델링에 필요한 다물체 요소 정보를 DB 로부터 동적으로 읽어 들이므로 쉽게 확장될 수 있다.

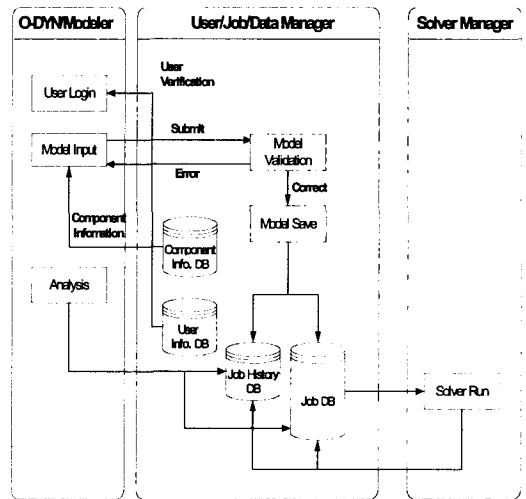


Fig. 3 O-DYN/Modeler data flow

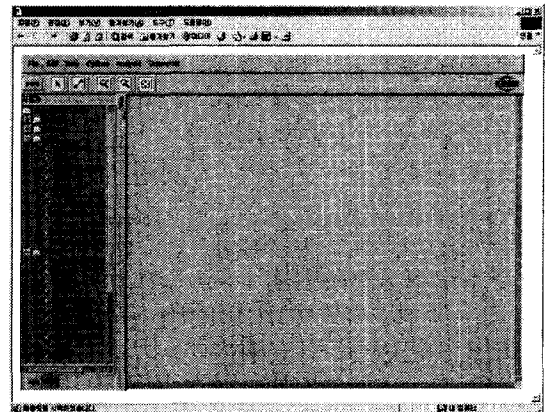


Fig. 4 O-DYN/Modeler

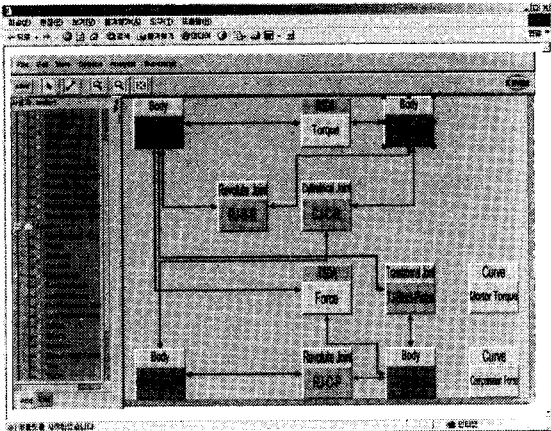


Fig. 5 Block-diagram type modeling in O-DYN/Modeler

### 2.4 JOB Control Module

JOB Control Module 은 PRE Module, POST Module, SOLVER Module 간의 상호작용을 제어하는 Job Manager, 사용자 인증을 담당하는 User Manager, 시스템의 모든 데이터를 DB 로 관리하는 Data Manager 가 JAVA Servlet 으로 구현되었다. Job Manager 는 PRE Module 에서 생성한 모델 데이터를 해석기 O-DYN/Solver 의 입력파일로 생성하여 SOLVER Module 에 전달하는 기능, SOLVER Module 에 해석을 요청하는 기능, SOLVER Module 의 해석 결과 파일을 DB 에 저장하는 기능, POST Module 이 요청한 시뮬레이션 결과를 전달하는 기능을 수행한다. User Manager 는 PRE Module, POST Module 의 사용자 인증요청에 따라 사용자 인증을 수행하는데 사용자 네트워크 주소, 사용자 id/password 를 이용한 2 단계 인증을 실시한다. Data Manager 는 PRE Module, POST Module, Job Manager 에서 사용되는 데이터를 DB 에서 읽거나 쓰는 기능을 수행하며, SOLVER Module 과의 인터페이스를 위해 DB 에 저장된 모델데이터를 파일로 쓰는 기능과 해석결과 파일을 읽어 들여 DB 에 저장하는 기능을 제공한다. Fig. 6 은 PRE Module 에서 저장된 모델 데이터를 JOB Control Module 이 SOLVER Module 에게 전달하여 시뮬레이션을 수행하고 시뮬레이션의 성공여부에 따라 사용자가 POST Module 을 이용하여 에러메시지나 결과를 분석하는 시스템의 작업 흐름과 각 단계에서 필요한 데이터 파일을 나타내고 있다.

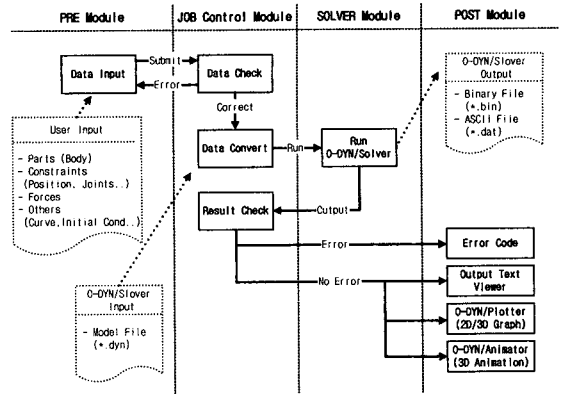


Fig. 6 Control flow of O-DYN system

### 2.5 SOLVER Module

SOLVER Module 은 다물체 계 동역학 해석기인 O-DYN/Solver 와 이를 구동시키는 Solver Manager 가 개발되었다. O-DYN/Solver 는 다물체 동역학 시뮬레이션을 위하여 다물체 동역학에 객체지향 모델링 및 프로그램 기법을 적용하여 C++로 개발하였다. Table 2 와 같이 현재 일반적으로 이용되는 물체(Body), 구속 조건(Constraints), 구속 조건을 이용한 조인트류, 힘 요소들을 라이브러리로 가지고 있다. Solver Manager 는 배치작업 형태로 해석기 O-DYN/Solver 를 구동시키며 JAVA Servlet 으로 구현되었다. O-DYN/Solver 에 대하여 간단히 설명하면 다음과 같다.

다물체 동역학 계의 운동방정식은 식(1)과 같이 정의된다.

$$\begin{bmatrix} \mathbf{M} & \Phi_q^T \\ \Phi_q & 0 \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}} \\ \lambda \end{Bmatrix} = \begin{Bmatrix} \mathbf{Q} \\ \gamma \end{Bmatrix} \quad (1)$$

여기서,

- $\mathbf{q}$ : position vector
- $\dot{\mathbf{q}}$ : velocity vector
- $\ddot{\mathbf{q}}$ : acceleration vector
- $\mathbf{M}$ : mass matrix
- $\Phi_q = [\partial\Phi_j / \partial q_i]_{m \times n}$ : constraint Jacobian matrix
- $\lambda$ : Lagrange multiplier
- $\mathbf{Q}$ : generalized force
- $\gamma$ : right side of constraint acceleration

이러한 다물체 동역학 시스템 운동방정식의 해법은 Haug<sup>17</sup>의 문헌을 참조 바란다. 이러한 운동방정식에 기초를 둔 상용 프로그램들은 모두 절차적 프로그래밍 기법에 기초를 두고 있다. 다물체 시스템의 동적 해를 구하기 위해서는 식(1)의 수치해석이 요구된다. 본 논문에서는 다물체계의 운동방정식 구성 및 수치해석도 클래스를 통하여 이루어진다. 본 논문에서는 식(1)의 구성 및 수치해석을 위하여 별도의 클래스를 정의하였고 동적 해석은 Fig. 7 과 같은 순서로 진행된다. 수치해석을 수행하는 클래스는 시스템 운동방정식을 구성하기 위한 벡터, 행렬, 수치해석 함수들을 가지고 있게 된다. Fig. 8 은 O-DYN/Solver 흐름도를 보여 주고 있다.

Table 2 O-DYN/Solver classes

Class type	Classes
Body	Rigid Body
Constraint	Ground, Position, Point, Revolute joint, Cylindrical joint, Translational joint, Bracket joint, Spherical joint, Rack-and-pinion, Gear joint, Driver
Force	Translational-spring-damper, Rotational-spring-damper, Beam, Bushing

제공한다. 또한 주석(annotation) 기능을 이용하여 시뮬레이션 결과의 토의가 가능하다.

2.6 POST Module

POST Module 는 2D/3D 그래프, 3D 애니메이션을 통한 시뮬레이션 결과 분석 및 가시화를 웹브라우저 상에서 제공하며 JAVA Applet 으로 개발되었다(Fig. 9). 2D/3D 그래프 형태의 결과분석을 제공하는 O-DYN/Plotter 는 다수 사용자간에 시뮬레이션 결과의 공유, 협업 및 정보교환이 가능하도록 개발되었으며 JWAVE 라이브러리를 사용하였다. 협업 및 정보공유를 위하여 사용자 A 가 시뮬레이션 결과 그래프의 특정 구역을 줌(zoom)하여 결과를 상세하게 본다면 인터넷을 통해 동일 시뮬레이션 결과를 보고 있는 사용자 B 에게도 같은 결과를

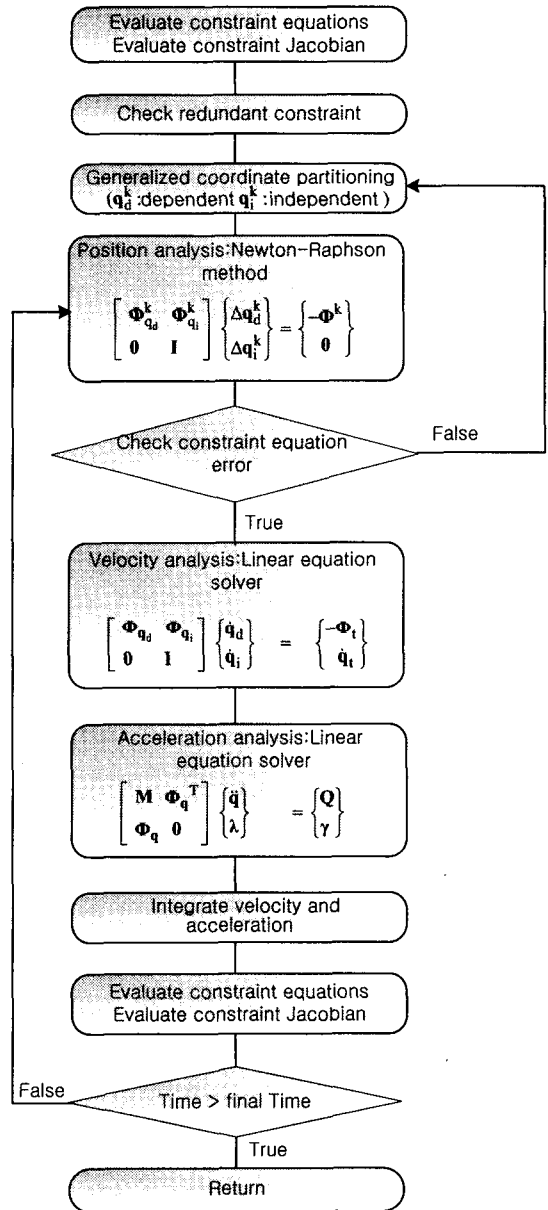


Fig. 7 O-DYN/Solver numerical analysis flow

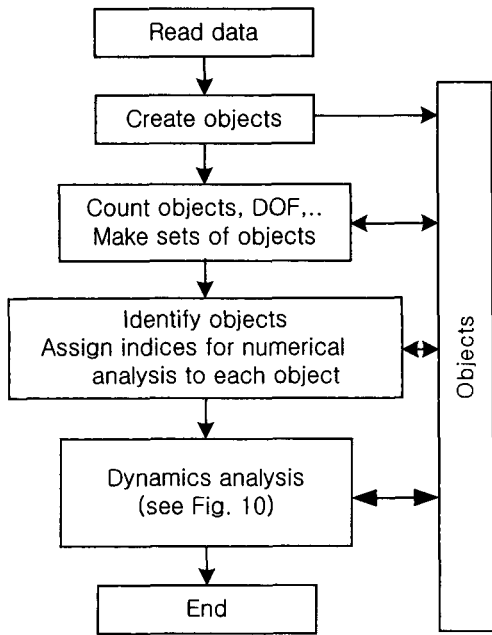


Fig. 8 O-DYN/Solver flow

다물체 동역학 해석 결과를 이용한 거동 가시화기능을 제공하는 O-DYN/ Animator 가 개발되었다. 본 논문에서는 웹 기반 시스템을 개발하기 때문에 사용자는 웹을 통하여 동적 해석 결과의 검토가 가능하고 다수의 동시 사용자 접속에 의한 협업을 지원하는 장점을 갖는다. 본 논문에서 개발한 O-DYN/Animator 는 3D 애니메이션을 통한 해석결과 분석 및 가시화를 웹 브라우저 상에 제공하며, OpenGL 기반의 JAVA OpenInventor 라이브러리를 사용하였다. 각 물체의 형상정보(Geometry) 는 iv 또는 VRML 형식을 이용한다. 형상정보와 O-DYN/Solver 의 시뮬레이션 결과인 동적 거동 정보(x, y, z,  $\theta_x$ ,  $\theta_y$ ,  $\theta_z$ )를 가지고 3차원 동적 거동을 애니메이션으로 가시화 할 수 있다. Fig. 10 은 O-DYN/Animator 의 시스템 구성도를 보여주고 있다. O-DYN/Animator 는 Importer 모듈, Creator 모듈 그리고 Viewer 모듈로 구성된다. Importer 모듈은 사용자가 CAD 로부터 생성한 iv 나 VRML 파일을 업로드(Upload)하여 형상정보를 생성하고, Creator 모듈은 사용자가 형상정보를 정의하여 생성할 수 있도록 한다. Viewer 모듈은 Importer 모듈과 Creator 모듈에서 만들어진 형상정보와 O-DYN/Solver 의

시뮬레이션 결과 파일을 토대로 3D 애니메이션으로 가시화 하는 기능을 갖는다. Fig. 11 는 해석 결과를 이용한 3D 애니메이션 예를 보여주고 있다. 화면 좌측에는 애니메이션 대상에 대한 모델 데이터를 나타내며 진한 색 블록이 애니메이션의 각 형상정보에 해당되는 것을 나타낸다. 예제의 경우 4 가지 형상정보를 가지고 있음을 알 수 있다.

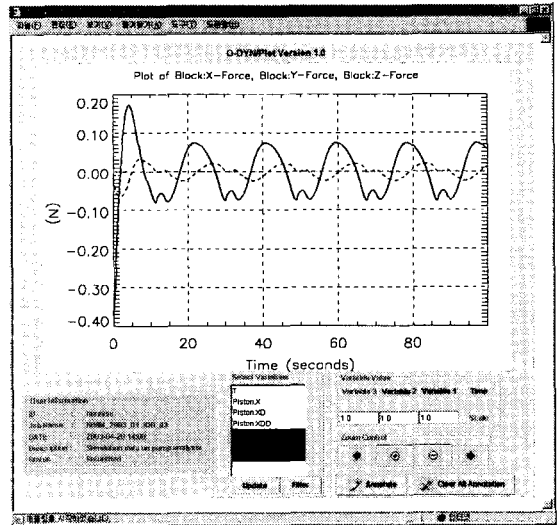


Fig. 9 O-DYN/Plotter

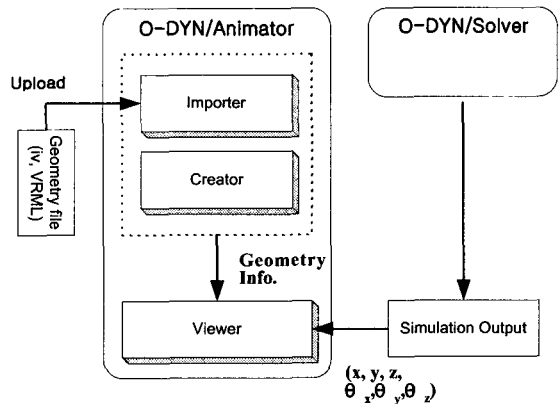


Fig. 10 O-DYN/Animator flow



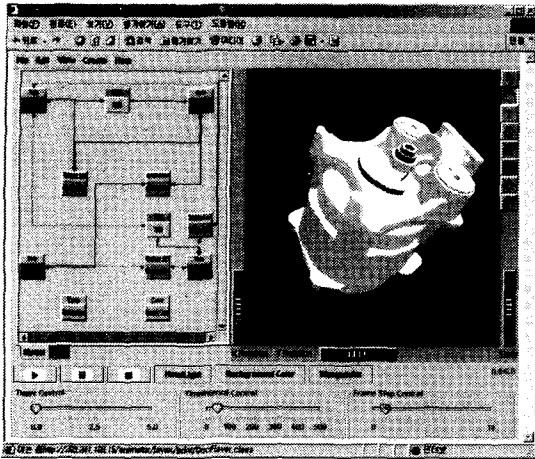


Fig. 11 O-DYN/Animator

### 3. 적용

본 논문에서 소개한 웹기반 다물체 동역학 시뮬레이션 시스템 O-DYN 을 이용하여 복동식 압축기의 동적 해석을 수행하였다. Fig. 12 에서 보여주는 복동식 압축기는 냉장고를 비롯한 냉각 장치에 많이 이용된다. 압축기의 고급화 경향에 따라 압축기의 진동 저감은 중요한 과제 중의 하나이다. 압축기의 진동 원인은 실린더의 압력 변화와 그에 따른 모터 토크의 변동에 주로 기인한다. 진동 저감을 달성하기 위해서는 냉매 압력과 모터 성능이 주어진 조건에서, 동적 거동과 그에 따른 진동을 야기하는 가진력 예측이 요구된다. 본 논문에서는 가진력을 예측하기 위하여 본체를 구속조건으로 고정하고 구속 조건에서의 반력을 가진력으로 평가한다. Fig. 13 은 압축기의 개략적인 형상과 입출력의 기준이 되는 좌표계를 보여주고 있다. Fig. 12 의 압축기에 대한 관성 특성과 다물체 동역학적 모델을 Table 3 과 Table 4 에서 각각 보여주고 있다. 압축기 모터의 토크 성능은 실험에 의하여 구해졌으며 Fig. 14 와 같다. 모터 토크는 회전자의 속도와 그에 따른 출력 토크를 Fig. 14 의 성능 곡선을 이용하여 정의하게 된다. 실린더 내에서의 압력은 회전자의 회전 각의 함수로 Fig. 15 와 같이 이론적으로 얻어진 특성 곡선을 이용한다. 압축기는 부품간의 접촉부에 회전 저항이 존재한다. 본 논문에서는 전체적인 거동에 중점을 두기 때문에 실험 결과를 기초로 하여 단지 회전자와 고정된 블

록(block)의 상대 회전 속도에 따른 감쇠로 정의하였다. 실험 결과에서의 모터 회전 속도와 회전 속도의 변동을 해석 모델과 비교하여 회전 저항으로 감쇠계수 2.5Nms/mm 를 택하였다. 본 압축기에 대한 O-DYN/Modeler 를 이용한 모델링은 Fig. 5 와 같다. Fig. 5 와 같이 Table 3, Table 4 의 다물체 모델링 요소들이 블록다이어그램으로 모델된다. 각 블록을 클릭하여 각 요소의 입력 파라미터 값을 설정한다. 그런 후 해석을 의뢰하면 O-DYN/Solver 에 의하여 동적 해석이 이루어진다. O-DYN/Solver 에 의하여 해석이 이루어지면 Fig. 9 의 O-DYN/Plotter 에 의하여 결과를 분석하게 된다. Fig. 16 은 피스톤의 위치를, Fig. 17 은 x 방향 반력을 O-DYN/Plotter 에 의하여 보여주고 있다. 전장에서 서술한 바와 같이 O-DYN/Plotter 를 이용하면 협업 지원이 가능하게 된다. 압축기의 위치는 주기적인 운동을 하며 반력은 회전자가 0°, 180°일때 최고 값을 갖으며 최대 크기는 약 70N 인 것을 알 수 있다. 이 반력은 압축기의 가진력으로 작용할 것이다. 본 논문에서는 시스템의 소개에 중점을 두기 때문에 압축기의 동적 해석 결과에 대하여는 간단히 소개한다. Fig. 11 은 거동을 O-DYN/Animator 에 의한 가시화를 보여주고 있다. 여기서 O-DYN/Plotter, O-DYN/Animator 는 다수의 사용자가 동일한 결과를 볼 수 있어 협업 지원이 가능하게 된다.

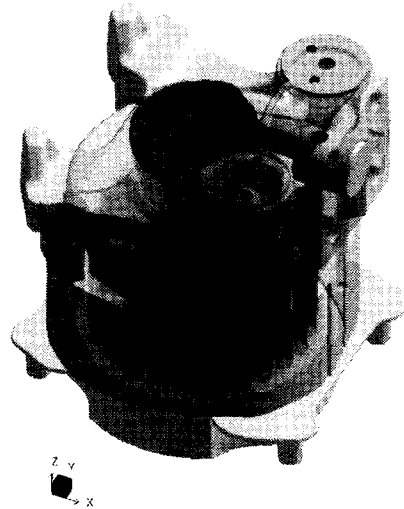


Fig. 12 Reciprocating compressor

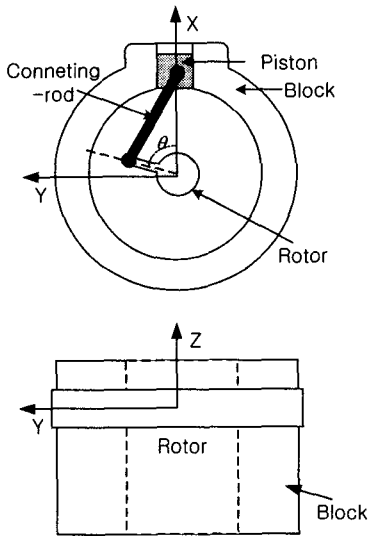


Fig. 13 Schematic diagram of the compressor

Table 3 Inertia properties of the compressor

Body	Mass(kg)	$I_{xx}$ (kgmm <sup>2</sup> )	$I_{yy}$	$I_{zz}$
Block	5.0	20.0	0.05	15.02
Rotor	0.04	500.0	500.0	30.0
Conneting-Rod	0.03	1.1	10.0	3.5
Piston	0.03	3.0	3.5	4.0

Table 4 Dynamic model of the compressor

Bodies	Block, Rotor Conneting-rod, Piston
Translational joint	Block-Piston
Revolute joint	Block-Rotor
Revolute joint	Conneting-rod-Piston
Cylindrical joint	Rotor-Conneting-rod
Motor torque	Fig. 14
Compression force	Fig. 15
Rolling resistance	Block-Rotor (2.5 Nms/mm)

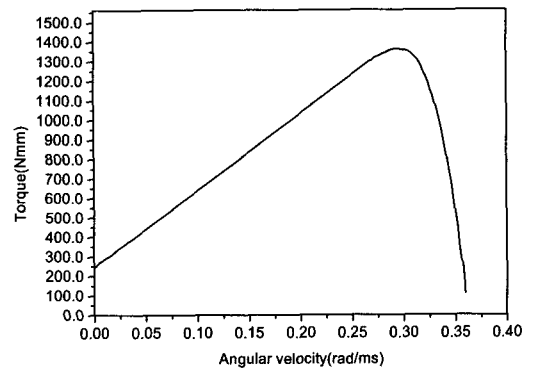


Fig. 14 Characteristics of motor

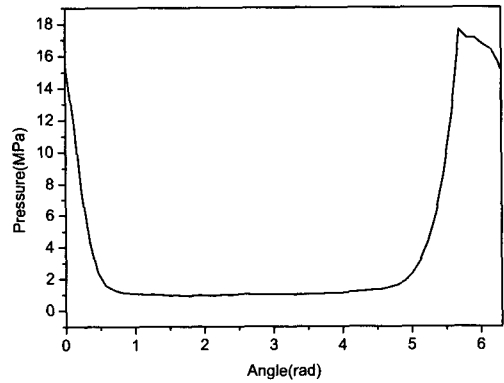


Fig. 15 Pressure versus rotor angle

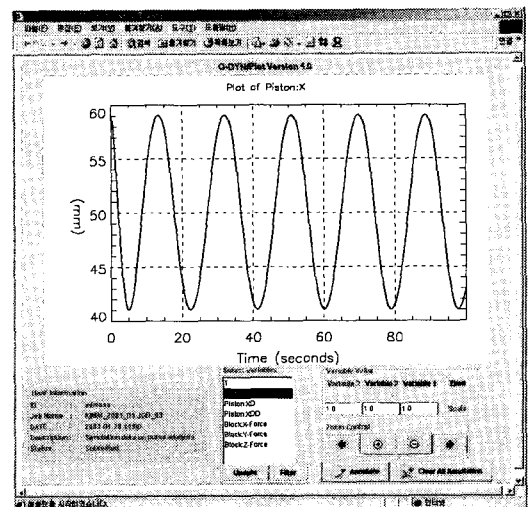


Fig. 16 x position of piston versus time

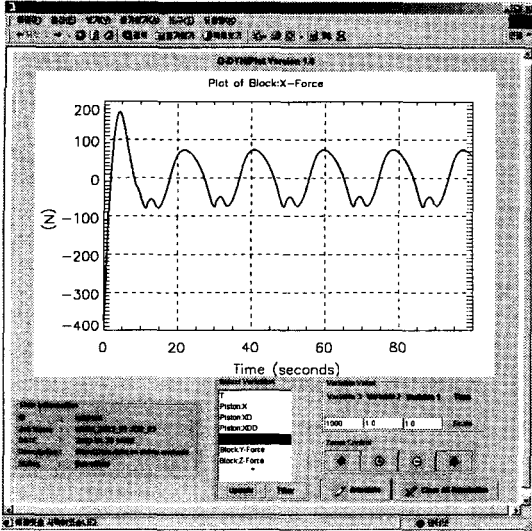


Fig. 17 x reaction force on block at origin

#### 4. 결론

본 논문에서는 웹기반 범용 다물체 동역학 시뮬레이션 시스템을 소개하였다. 시뮬레이션 시스템의 모든 사용자 인터페이스는 웹 브라우저와 인터넷을 통하여 이루어진다. 동적 시뮬레이션에 필요한 모든 수치계산은 웹서버에서 수행된다. 웹에서의 모든 사용자 인터페이스는 JAVA Applet 에 기반을 두고 개발되었다. 결과적으로 사용자는 사용자 컴퓨터 환경과 위치에 상관없이 인터넷과 웹 브라우저를 통하여 다물체 시스템의 동적 시뮬레이션 수행이 가능하다. 또한 동시적인 정보의 공유로 협업 및 동시공학 구현이 가능하고 정보의 데이터베이스를 이용한 체계적인 관리가 가능한 장점이 있다.

#### 후 기

본 연구는 과학기술부에서 지원하는 주력산업의 고부가가치화 사업 “웹기반 범용 동력전달계 성능해석 시스템 개발”과제의 연구비로 수행되었음을 밝히며 본 연구의 지원에 대하여 감사 드립니다.

#### 참고문헌

1. Lee, Y. H., Kwak, S. G. and Kim, S. H., “Web-Based Simulation under Distributed Environment,” Journal of the Korea Society for Simulation, Vol. 7, No. 2, pp. 79-90, 1998.
2. Kim, K. Y. and Nam, Y. H., “A Visual Modeling Environment for Web-based Simulation,” Journal of the Korea Society for Simulation, Vol. 8, No. 1, pp. 101-111, 1999.
3. Jin, G. J., Kwak, M. K. and Heo, S., “Development of Web-Based Engineering Calculation Program Using JavaScript,” Proc. of KSNVE Spring Conf., pp. 411-416, 2002.
4. Seo, J. W. and Kang, S. K., “A Study on the Fault Diagnosis in Web-based Virtual Machine,” Proc. of KSPE Spring Conf., pp. 430-434, 2001.
5. Kim, D. H., Kim, S. H., Kim, J. H. and Han, K. S., “A Study on Web based Monitoring System of Machine Tool,” Proc. of KSPE Autumn Conf., pp. 60-63, 2001.
6. Shen, W. and Ghenniwa, H. H. “A Distributed Multidisciplinary Design Optimization Framework Based On Web And Agents,” Proc. of ASME DETC2002, CD-ROM, 2002.
7. Knoth, J., “Engineering via the Internet,” Computer-Aided Engineering, pp. 54-60, 1996.
8. <http://www.lmsintl.com/>
9. <http://www.adams.com/>
10. <http://www.functionbay.com/>
11. <http://www.simpack.de/>
12. <http://java.sun.com/>
13. <http://www.tgs.com/>
14. <http://www.vni.com/>
15. <http://www.ilog.com/>
16. <http://www.caucho.com/resin/>
17. Haug, E. J., Computer Aided Kinematics and Dynamics of Mechanical System, Allyn and Bacon, USA, 1989.