

월드와이드웹용 물리교육 코스웨어 개발에 관한 연구

부동혁 * · 박규은 ** · 김규용 **

A Study of Physics Courseware Development for World Wide Web

Bu, Dong-Hyuk · Park, Kyu-Eun · Kim, Kyu-Yong

Abstract

Many kinds of material have been researched and analyzed to get the essential guidelines for developing physics CAI courseware which is capable of using in WWW. It has turned out that it enables to enhance the courseware quality and productivity to use the following functions such as communication, presentation, branching and interactivity basically provided by web environment. It has also turned out it is necessary to use the distributed processing like JAVA for physics simulation courseware.

JAVA applet, which simulates projectile motion, has been made and inserted in the physics home page. The experimentation results through WWW are following:

1. In spite of the small file size of 14 kbyte, it makes animation processing and video/sound production much easier.

* 제주동중학교 교사

** 제주대학교 사범대학 과학교육과 교수

2. JAVA applet enables it possible for the learners to interact with simulator.
3. It shows just 1% below performance error. Those results show that the JAVA applet is practical and effective enough for physics class.

I. 서 론

교육적으로 그 응용 가치가 높다고 여겨지는 인터넷(Internet)의 여러 서비스들 중에서 월드와이드웹(WWW: World Wide Web, 이하 웹으로 약칭)은 쉬운 사용자 인터페이스와 자원 제작의 용이성으로 인하여 중요한 역할을 할것으로 기대된다. 최근 몇 년 동안 폭발적으로 성장한 웹은 인터넷을 대중화 시키는데 큰 역할을 하였으며, 교육적 활용에 관한 연구도 광범위하게 이루어지고 있다. 웹이 가지고 있는 멀티미디어적 특성과 계속 확장되고 있는 기능들, 그리고 풍부한 자원과 지원 기술의 발전 추세를 감안할 때 CAI 코스웨어에서도 웹이 폭넓게 응용되리라는 것은 쉽게 예상되는 바이다. 실제로 세계 각국에는 웹의 교육적 특성을 일반화시키기 위해 많은 노력을 기울이고 있다. 미국은 정보기반구축사업인 NII(National Information Infrastructure)의 교육적 응용분야로서 여러개의 하부 프로젝트를 수행하고 있다. DLT(Digital Library Technology)계획¹⁾, RSD(Public Use of Remote Sensing Data)계획²⁾은 그 대표적인 예로서 웹이 중요한 수단으로서 활용되고 있다. 우리나라에서도 교육부가 1996년 9월에 종합교육정보 서비스인 에듀넷(EduNet)을 출범시켰다. 그리고 교수·학습용 멀티미디어 데이터베이스와 학술정보 데이터베이스를 구축하고 원격교육의 확대를 꾀하는 등 인터넷과 웹의 교육적 응용에 대한 노력이 지속적으로 이루어지고 있다.

그러나 웹의 교육적 잠재력과 풍부한 자원은 CAI 코스웨어 등과 같이 조직화된 체계를 통해서 활용될 때 그 가치를 더할 수 있을 것이다. 물리교육 CAI 코스웨어와 같이 비교적 높은 수준의 기술적 기교를 필요로 하는 프로그램을 웹에서 구현하기 위해서는 웹의 구체적 특성을 충분히 고려한 기술 개발이 선행되어야 한다. 그리고 이들 기술이 CAI 코스웨어 제작에 활발하게 응용될 수 있도록 일반화 시킬 필요가 있다. 더구나 컴퓨터와 통신 관련 분야 기술이 급격히 발전하고 있으며 응용 장비들의 교체 주기가 매우 짧아진 것을 감안하면, 컴퓨터통신에 관한 물리적 자원이 투입됨과 동시에 이를 이용하는 프로그램과 활용 방법도 같이 제공되어 낭비적 요소가 발생하지 않도록 해야 한다. 따라서 인터넷

이라는 새로운 매체를 기반으로 한 물리교육 CAI 코스웨어 개발에 관하여 분야별 기술을 신속하게 확립할 필요성이 있으며, 웹을 포함한 인터넷 자원을 물리교육에 활용하기 위한 연구가 폭넓게 수행되어 정보화시대의 과학교육 장면에 대처해 나가는 것이 중요하다고 본다.

그러므로 본 연구에서는 중등학교 물리교육 CAI 코스웨어를 개발하는데 활용할 수 있는 웹의 자원과 문제점을 분석·정리하여 물리교육 CAI 코스웨어 개발에 관한 지침을 얻고, 이를 기초로 하여 물리교육 CAI 코스웨어에 활용될 수 있는 예시적인 프로그램을 제작함으로써 웹용 물리교육 CAI 코스웨어를 구현하기 위한 기술적 방법을 탐색하고자 한다.

II. 기본 이론

1. 월드와이드웹

웹은 CERN의 T.Beners-Lee가 처음 개발한 하이퍼미디어(Hypermedia) 방식의 대규모 정보 서비스 시스템으로서, 1990년 발표된 이래 NCSA(National Computer Security Association)의 모자이크(Mosaic)과 같은 멀티미디어 중심의 웹브라우저의 보급에 힘입어 인터넷을 중심으로 급속히 성장하였다. 기존의 하이퍼미디어 시스템들이 사용자 접속 도구나 편집 기술 및 자료 저장 기법에 치중하여 개발된 반면, 웹은 전세계에 광범위하게 퍼져있는 여러 형식의 자료를 정보 서비스 환경에 접목시키는 데 초점을 두었다. 이를 위하여 다음과 같은 근본적인 문제를 해결하고 있다.³⁾

- 하이퍼미디어 문서를 기술하는 언어
- 망을 통해 하이퍼미디어 문서를 전송하는 데 필요한 통신 규약
- 인터넷을 통해 전 세계에 퍼져있는 자료를 식별하기 위한 기법

위의 세가지는 각각 HTML(Hyper-Text Markup Language) 와 URL(Uniform Resource Locator) 그리고 HTTP(Hyper-Text Transfer Protocol)에 대응되며 웹의 성격을 가장 잘 나타낼 수 있는 요소이다.⁴⁾

2. 웹을 위한 CAI 코스웨어의 제작 환경

HTML 규격은 문서의 장식,분기,입력양식을 통한 상호작용 등을 지원하지만 고도의 연산과 다양한 상호작용 그리고 애니메이션이 수반되는 과학 시뮬레이션 코스웨어를 구현

하는 도구로서는 턱없이 기능이 부족하다. 그리고 웹에서 주로 사용하는 HTTP는 단속적인 프로토콜(stateless protocol)로서 통신회선의 부하를 경감하는 등 여러 가지 긍정적인 측면을 갖고 있음에도 불구하고, 지속적인 세션(session)이 필요한 기능의 구현에는 제한 요소로서 작용한다.

1) 학습의 관리의 어려움

교육용 소프트웨어는 학습자의 능력, 학습 경력과 같은 배경을 염두에 두고 학습 내용을 제시하고 반응할 수 있어야 하며, 어떤 개념에 어려움을 느끼는 학습자에게 적절한 처방을 제공할 수 있어야 한다. 이를 위해서는 코스웨어가 학습자의 상태와 행동을 계속 추적하고 측정된 사항들을 기록해 두어야 할 필요성이 생기는데, 이는 동적인 프로토콜을 사용하는 웹의 특성과는 상치되는 면이다. B.Ibrahim⁴⁾은 이러한 문제점을 그림1)과 같이 CGI 프로그램을 이용하여 해결할 것을 제안하였다.

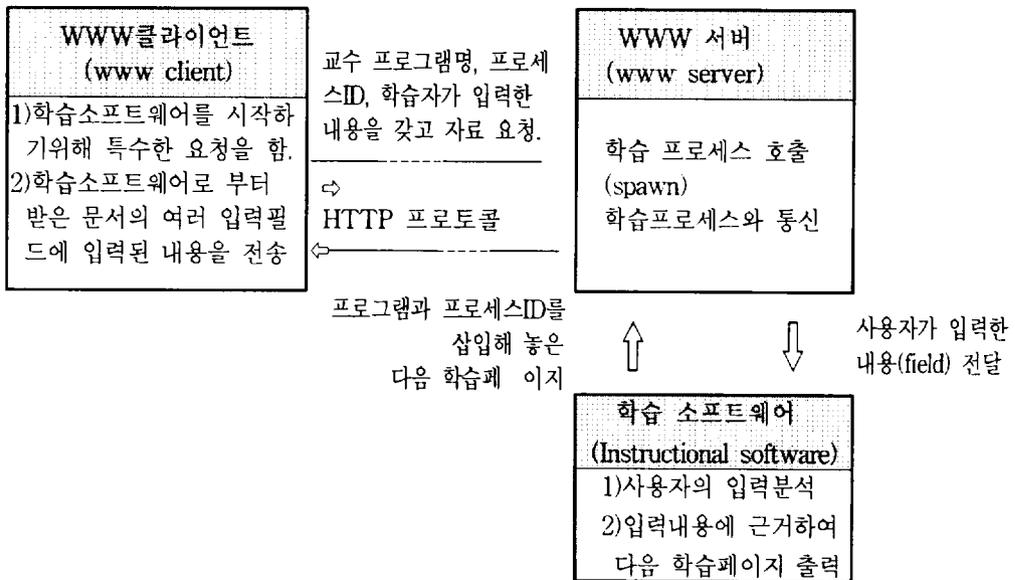


그림 1. WWW를 통한 학습 소프트웨어와 원격 사용자 사이의 상호작용 양식

그러나 이러한 접근은 몇가지 난점을 갖고 있다. 첫째는 동시에 여러 사용자로부터 자료 요청이 발생할 때 서버에 과부하가 걸리기 쉽다는 것이고 둘째는 학습자와의 상호작용이 HTML의 문법에 의해 제한받는다라는 점이다. 세번째로 들 수 있는 것은 과거 상태의 재현에 관한 문제이다. 즉 지금까지의 모든 상황의 변화를 기록해 두고 다음 연결에서 처

음 부터 다시 재연출하는데 사용해야한다는 점이다. 이러한 방법은 컴퓨터의 성능을 상태의 재현에 소모하며 네트워크의 통신량을 증가시킬 뿐만 아니라 연결이 도중에 끊어졌을 경우, 상태가 파괴된다는 문제점을 안고 있음을 Martin Gleeson⁷⁾과 이택경⁵⁾은 지적하고 있다.

2) 상호작용에 관한 제한점

여기서 말하는 상호작용은 학습자가 하이퍼텍스트의 링크를 선택하여 이미 작성된 또 다른 내용을 열람하는 수준의 소극적인 것을 뜻하는 것이 아니다. 학습자가 마우스를 조작하거나 문장이나 수치를 입력하면 그 변화가 동적으로 생성되고 학습자에게 피드백을 줄 수 있는 보다 활발한 상호작용을 의미한다. J.Newmarch⁶⁾, Martin Gleeson과 Tina Westaway, T.Martin⁷⁾, B.Ibrahim⁸⁾은 웹 환경에서 HTML의 입력문(forms)과 CGI 프로그램을 이용하여 어느 정도의 상호작용을 실현하는 것이 가능하나 제한 사항이 많음을 그들의 연구에서 밝히고 있다.

CGI 프로그램의 URL을 요청받은 서버는 해당 프로세스를 실행하며 그동안 클라이언트의 사용자는 개입할 수 없다. 그리고 처리 결과를 HTML 문서 등의 형식으로 클라이언트에 전송하고 나면 입력문이나 이미지맵(image-map)과 같은 정해진 방법에 의하여 다시 요청이 올 때까지 클라이언트와 연결이 해제된다. 따라서 클라이언트의 학습자가 마우스 커서를 이리저리 옮겨다니면 그림이 변한다거나, 포사체의 운동을 애니메이션으로 표현하는 시뮬레이션과 같은 것을 구현하는 것이 어렵다. 즉, 코스웨어의 내용에 제약을 크게 받을 수 밖에 없다.

3) 서버의 과부하 문제

D.Wojtowicz, R.Wilhelmson, M.Ramamurthy⁹⁾는 상호작용적 영상을 위한 클라이언트 환경에 관한 연구에서 화상의 처리와 같이 단위 요구당 처리량이 많을 때 네트워크상의 서버의 능력은 쉽게 포화되며, 이를 해결하기 위해 서버의 능력을 증가시키는 것은 단순한 문제가 아니라는 점을 지적하고 분산처리의 필요성을 강조하였다. 즉 증가하는 사용자의 수에 맞추어 서버를 증강하는 것은 어려우나 클라이언트 시스템의 수는 사용자와 거의 같은 비율로 증가하는데 착안하여 해결책을 모색하였다.

3. 분산처리 지향 언어의 도입

이상의 논의에 비추어 볼 때, 물리교육용 CAI 프로그램과 같이 복잡한 계산이나 변화가 많은 애니메이션을 요구하는 프로세스를 소화하기 위해서는 클라이언트측 역할 증대가 필요함을 알 수 있다. 복잡한 처리를 CGI 프로그램과 같은 서버측 실행환경에 일임하는

것은 과부하 등의 여러 가지 문제가 수반되고, HTML 형식은 근본적으로 한계를 갖고 있기 때문이다. 이런 문제점을 해결하는 대안으로서 최근에 개발된 자바(JAVA)와 같은 클라이언트/서버 지향 언어의 도입을 고려해볼만 하다. 이러한 언어는 CAI 프로그램이 요구하는 복잡한 처리를 클라이언트 측에서 수행할 수 있는 능력을 갖고 있기 때문이다.

1) 자바언어의 특징

자바는 선마이크로시스템사(Sun Microsystems Inc.)에서 C++를 모태로 하여 개발된 개체지향 언어로서(1996년 2월에 정식버전 발표) C++와 형식이 매우 유사하다. 이 언어의 제작자인 J.Gosling과 H.McGilton은 자바언어백서¹⁰⁾에서 자바의 특징을 다음과 같이 요약하고 있다.

- 이식성이 강한 인터프리터이다.
- 애플릿 형태로 웹에서 활용될 수 있다.
- 멀티쓰레딩 지원
- 네트워크 기능의 지원
- 데이터베이스 연결성 지원

2) 자바언어의 단점

전술한 여러 가지 장점에도 불구하고 자바 언어는 다음과 같은 단점을 가지고 있다. 우선 C나 C++와 같이 특정 기계에 최적화된 언어로 작성된 응용 프로그램에 비하여 자바의 프로그램은 수행 속도가 느리다. 그리고 애플릿의 로딩(loading)과 실행이 불규칙하여 안정성이 다소 부족하며, 프로그램에 능숙하지 않은 사람들에게 개체지향과 멀티쓰레딩 개념은 결코 쉬운 것이 아니다. 또한 이식성을 큰 장점으로 내세우는 것에 비하여 실제 자바 응용프로그램이나 애플릿이 수행될 수 있는 플랫폼은 많지 않은 형편이다. 현재 자바 애플릿을 사용할 수 있는 클라이언트의 운영체제는 윈도우즈3.1, 윈도우즈95, SunOS 그리고 매킨토시 정도이다. 그리고 자바 애플릿은 클라이언트 측에서 실행되므로 서버와 통신 선로의 부하를 줄인다는 장점이 있는 반면에 서버측에서 관리해야할 성질의 정보를 취급하는데 취약하거나 비교적 복잡한 과정을 거쳐야 한다. 웹서버나 웹브라우저는 보안 체계를 갖추고 있어서 네트워크 건너편의 파일을 조작하는데 제한을 두기 때문이다. 따라서 접속자의 관리나 학습자료 데이터베이스의 운영 등의 중앙 집중형 프로세스들은 아무래도 서버측 실행 환경에서 담당하는 것이 자연스럽다고 판단된다.

III. 예시적인 물리 시뮬레이션 애플릿의 제작

웹 환경이 가진 자원이 매우 풍부하며 자바와 같은 언어가 병용될 때 그 효용가치가 더욱 커질 것으로 판단되므로 그 응용에 대한 연구를 계속할 필요가 있다고 본다. 이에 본 연구에서는 예시적인 물리실험용 자바 애플릿을 개발하고 그 과정에서 발생하는 문제점을 파악함과 동시에 결과물의 성능을 조사하여 더 발전적인 과제를 수행하는데 유용한 자료를 얻고자 하였다.

1. 자바 애플릿 '포사체 운동 실험기'의 개요

본 애플릿은 포사체운동을 시뮬레이션하는 기능을 가지고 있어서, HTML 형식의 문서에 삽입하여 웹용 물리교육 코스웨어를 제작하는데 활용할 수 있도록 고안되었다. 학습자가 대포와 표적의 위치, 발사 속도와 각도, 중력, 저항력, 포탄 질량의 초기값을 설정한 후, 발사 명령을 주면 그에 따른 포사체의 애니메이션이 연출된다. 학습자는 이 애플릿을 사용하여 여러 가지 조건에서 포사체 운동을 실험하고 그 결과를 측정하여 각 물리 변수들과 포사체 운동의 관계를 도출해낼 수 있어야 한다.

2. 개발 방향

시뮬레이션형 물리교육 CAI 코스웨어에서 필요한 요소들을 자바애플릿에 포함시켜 제작함으로써 다음에 관련된 자료들을 얻을 수 있도록 하였다.

- 연산에 의한 애니메이션
- 고색상의 화상 출력
- 샘플링된 효과 음향 재생
- 상호작용 기능
- 파일 용량

3. 제작에 사용된 하드웨어와 소프트웨어

컴파일러: JAVA (Sun Microsystems, Inc. 제품), HTML3.0

운영체제: Microsoft Windows95

브라우저: Netscape Navigator2.0-3.0, Microsoft Internet Explorer2.0-3.0

서버: Microsoft FrontPage1.1 Beta, Netscape Commerce, NCSA HTTPd1.4

표 1. '포사체 운동 실험기' 제작 과정에서 사용한 장비 및 네트워크 자원

구분	명칭	사양
컴퓨터	서버용	개인용 컴퓨터(CPU: 펜티엄586 120MHz) 1대 워크스테이션(교육청 전산실) 1대
	클라이언트용	개인용 컴퓨터(CPU: 펜티엄586 100/150MHz) 3대
네트워크 자원	랜카드	3c509 combo (3com제품)
	모뎀	28,800 bps
	LAN	개발용 컴퓨터를 제주도교육청 전산실 LAN의 허브에 접속 (Ethernet: 10BASET케이블, 10M bps)
	프로토콜	TCP/IP,PPP(가정에서 모뎀 접속시)
	클라이언트 IP주소	203.234.44.251, 203.234.44.252 동적 주소(가정에서 PPP 접속시)
	서버의 IP 주소	203.234.44.253 203.234.44.2 (ns.ed.cheju.kr/www.ed.cheju.kr) 168.126.3.3 (soback.kornet.nm.kr)
	인터넷 접속 방법	LAN의 호스트에서 교육망을 통한 접속 가정에서 모뎀을 통한 KORNET의 접속 서비스

4. 자바 애플릿 프로그램 작성

1) 사례 분석

월드와이드웹 사이트에 있는 실험적인 애플릿들을 수집하고 소스를 분석하였다. 본 연구의 목적에 가장 근접한 것은 미국 오레곤대학의 자바연구회 홈페이지(URL: <http://jersey.uoregon.edu/vlab/>)에서 운영하는 가상실험실(Virtual Lab)의 애플릿들이었다. 특히 Sean Russell이 제작한 'Fire Away'에서 개발하려는 애플릿의 기본 틀을 얻을 수 있었다. 그러나 이 애플릿은 다음과 같이 몇가지 미흡한 점을 안고 있어서 실제 물리 수업에 활용되기 위해서는 개선이 필요하였다.

- 실행상의 안정성이 부족하다. 접속 상태에 따라 잘 실행되지 않는 경우가 있으며 재 접속(reload) 했을 때에는 프로그램이 파괴된다.
- 사용자 인터페이스가 산만하며 불안정하며 정량적인 측정이 불가능하다.
- 그래픽 처리가 미약하며 안정감이 떨어진다. 그림 조각들의 위치가 임의로 변하고 화면 스크롤과 같은 상황에 대한 대비가 부족하다.

이 애플릿은 인터넷에서 활용될 수 있는 시뮬레이션형 물리교육 CAI코스웨어의 방향을 잘 제시하고 있으나, 실제로 물리 학습 장면에 활용하는데에는 부적절한 것으로 판단되었다.

2) 프로그램의 설계

‘포사체 운동 실험기’는 두 개의 클래스로 구성되며 구조와 기능은 다음과 같다.

(1) 실험기 클래스

① 클래스명 - Fire.class

② 애플릿 전체의 틀을 구성하는 주된 클래스로서 다음과 같은 작용을 한다.

- 애플릿을 시동하고 소멸시킨다.
- 물리 변수 조작을 위한 조정판 개체를 생성하고 가동시킨다.
- 애니메이션이 이루어지는 그림판 개체를 생성하고 가동시킨다.
- 그림과 음향 파일을 읽어 개체를 생성한다.
- 각 개체의 속성을 초기화한다.
- 조정판에서 이루어진 조작을 받아들이고 이에 반응한다.

③ 탄도 연산 메소드(methode)

메소드 DrawBullet(Graphics)은 포탄의 운동을 시뮬레이션하고 효과 음향을 재생하는 핵심 메소드이다. 포사체 운동의 시뮬레이션을 위해 다음 순서를 따라서 식을 전개하였다.

포사체에 작용하는 외력은 다음 식으로 구할 수 있다. 여기서 \vec{F} 는 외력, \vec{W} 는 중력, \vec{f} 는 공기의 저항력을 나타낸다.

$$\vec{F} = \vec{W} + \vec{f} \quad \text{-----(1)}$$

이어서 포탄 속도의 x,y 성분인 v_x, v_y 로부터 포탄의 운동각 θ 를 구한다.

$$\theta = \tan^{-1}\left(\frac{v_y}{v_x}\right) \quad \text{-----(2)}$$

포탄에 작용하는 중력과 공기의 저항력은 다음 식과 같다. m은 포탄의 질량, g는 중력 가속도, \vec{j} 는 y방향의 단위 벡터이며, 공기의 저항력은 속도 \vec{v} 의 크기의 제곱에 비례하는 것으로 하였다. 여기서 β 는 공기의 저항계수이다.

$$\vec{W} = -mg\vec{j} \quad \text{-----(3)}$$

$$\vec{f} = -\beta v^2 \frac{\vec{v}}{|\vec{v}|} \quad \text{-----(4)}$$

(2)식에서 구한 각 θ 를 이용하여 저항력의 x,y 축 성분 f_x, f_y 를 구하고, 이어서 (1),(3),(4)식을 이용하여 포탄에 작용하는 외력의 x,y 축 성분 F_x, F_y 를 구하면 (5)식을 얻

을 수 있다.

$$\begin{aligned} f_x &= f \cos \theta & f_y &= f \sin \theta \\ F_x &= f \cos \theta & F_y &= -mg + f \sin \theta \quad \text{-----(5)} \end{aligned}$$

(5)식의 양변에 dt 를 곱하여 다음과 같은 식을 얻는다.

$$dv_x = \frac{f}{m} \cos \theta dt \quad dv_y = \frac{f}{m} \sin \theta dt - g dt \quad \text{--(6)}$$

그런데, 어떤 시간 간격 Δt 동안 속력의 증분을 구하려면 위의 식을 시간 간격 Δt 에 걸쳐서 적분을 해야 한다. 그리고 속력의 증분을 현재 속력에 더하면 Δt 후의 속력을 얻을 수 있을 것이다. 그러나 시간 간격이 매우 작을 때에는 $\Delta t \approx dt$ 라 볼 수 있으므로, (6)식으로부터 다음과 같이 속력의 증분에 관한 근사식을 얻을 수 있다.

$$\Delta v_x \approx \frac{f}{m} \cos \theta \Delta t \quad \Delta v_y \approx \frac{f}{m} \sin \theta \Delta t - g \Delta t \quad \text{--(7)}$$

이제 시간 간격 Δt 후의 속력 v_x', v_y' 은 다음과 같이 쓸 수 있다. 여기서 v_x, v_y 는 이전 속력, $\Delta v_x, \Delta v_y$ 는 속력의 증분을 x, y 축 성분별로 각각 나타낸 것이다.

$$v_x' \approx v_x + \Delta v_x \quad v_y' \approx v_y + \Delta v_y \quad \text{----(8)}$$

한편, 이전 포탄의 좌표를 x, y 라 하고 시간 간격 Δt 후의 새 좌표를 x', y' 이라 할 때, 다음과 같이 쓸 수 있다. 여기서 $\Delta x, \Delta y$ 는 Δt 후의 위치 증분이다.

$$x' = x + \Delta x \quad y' = y + \Delta y \quad \text{--(9)}$$

시간 Δt 후의 위치의 증분을 얻으려면 속력 함수를 시간 Δt 에 걸쳐 적분해야 한다. 그러나 시간 간격이 매우 작을 때, 속력은 거의 변화가 없고 등속 운동에 가까워진다. 다음 식은 (8)식에서 얻은 순간 속력에 시간 Δt 를 곱하여 위치 증분의 근사치를 얻은 것이다.

$$\Delta x \approx v_x' \Delta t \quad \Delta y \approx v_y' \Delta t \quad \text{--(10)}$$

따라서 속도와 위치에 관한 (8)식과 (9)식은 각각 다음과 같이 표현될 수 있다.

$$v_x' \approx v_x + \frac{f}{m} \cos \theta \Delta t \quad v_y' \approx v_y + \frac{f}{m} \sin \theta \Delta t - g \Delta t \quad \text{--(11)}$$

$$x' \approx x + v_x' \Delta t \quad y' \approx y + v_y' \Delta t \quad \text{--(12)}$$

(12)식은 실제 프로그램에서 포탄의 위치를 결정하기 위해 사용되었는데, 이전 위치로

부터 새로운 위치를 구해나가는 방법을 택하고 있다. 학습자는 변수 조정판에서 '시간(dt)'을 조절하여 Δt 의 값을 설정하고, 점이 찍히는 간격을 조정할 수 있다. 근사식을 사용하므로 포탄의 위치에는 오차가 포함되는데, Δt 의 값을 0.1초로 할 때, 포사체 운동 애니메이션은 불과 1% 이내의 오차를 나타내었다. 따라서 컴퓨터 모니터상에서의 가상 실험을 전제로 하는 본 애플릿의 목적상 실용성은 충분하다고 판단된다. 결국 실험 결과의 정밀도를 다소 희생하고 프로그램의 논리를 단순화시킴으로써, 애플릿 크기를 작게하고 실행 속도를 향상시키는 결과를 얻을 수 있었다.

(2) 조정판 클래스

① 클래스명 - ControlPanel.class

② 변수조정판을 나타내는 클래스로 다음 작용을 한다.

- 실험기 클래스(Fire.class)에 의해서 생성된다.
- 사용자의 조작을 받아들이고 적당한 수치로 변환시켜 실험기 클래스로 전달한다.
- 발사 메시지를 실험기 클래스에 전달한다. 실제 변수값의 변경이나 포탄발사 등의 행동은 이 클래스에서 이루어지지 않는다.

5. '포사체 운동 실험기' 애플릿의 실행

1) HTML 문서의 작성

본 애플릿의 성능 실험을 위하여 예시적인 HTML 문서를 제작하고 다음과 같이 애플릿을 삽입하였다.

```
<APPLET code="Fire/Fire.class" width="650" height="450" align="center">
```

2) 실행 결과

그림4)와 같이 세계의 서버에 '포사체 운동 실험기' 애플릿을 삽입한 HTML 문서를 적재하고, 세 개의 클라이언트에서 웹 브라우저를 가동하여 LAN, 전용선, 전화선을 통하여 접속을 시도하였다. 접속 실험에 사용된 웹서버 프로그램은 가장 널리 쓰이고 있는 NCSA의 HTTPd/1.4 와 넷스케이프사의 Commerce 서버이며 두 서버 프로그램은 유닉스에서 가동되고 있다. 그리고 이 보다는 소규모의 서버인 마이크로소프트사의 FrontPage 1.0을 LAN에 접속된 개인용 컴퓨터에서 가동시켰다. 웹 접속에 사용된 브라우저는 가장 일반적으로 쓰이는 넷스케이프사의 Navigator3.0과 마이크로소프트사의 Explorer3.0이며, LAN에 접속된 두 개의 개인용 컴퓨터와 가정의 개인용 컴퓨터에 설치하여 사용하였다.

(1) '포사체 운동 실험기' 애플릿의 로딩(loading)

각 웹서버와 클라이언트들간의 네트워크상의 거리 및 조건이 동일하지 않으므로 웹서

버 프로그램과 서버 컴퓨터의 종류 및 운영체제에 따른 로딩 시간의 차이점을 비교할 수 없었다. 단 203.234.44.253(HTTPd: FrontPage)과 203.234.44.2(HTTPd: Commerce서버)는 LAN상에서 근접한 거리에 있어 거의 동일한 조건을 가지게 되므로 비교가 가능하였다.

① 서버 컴퓨터의 종류에 따른 결과

PC급인 서버(203.234.44.253)보다 워크스테이션급의 서버(203.234.44.2)에서 로딩 및 가동 속도가 더 빨랐으나 큰 차이는 보이지 않았다. 각 서버 프로그램이 특정한 운영체제를 요구하므로 서버프로그램의 종류나 운영체제에 따른 성능차이를 독립적으로 비교하는 것은 불가능하였다.

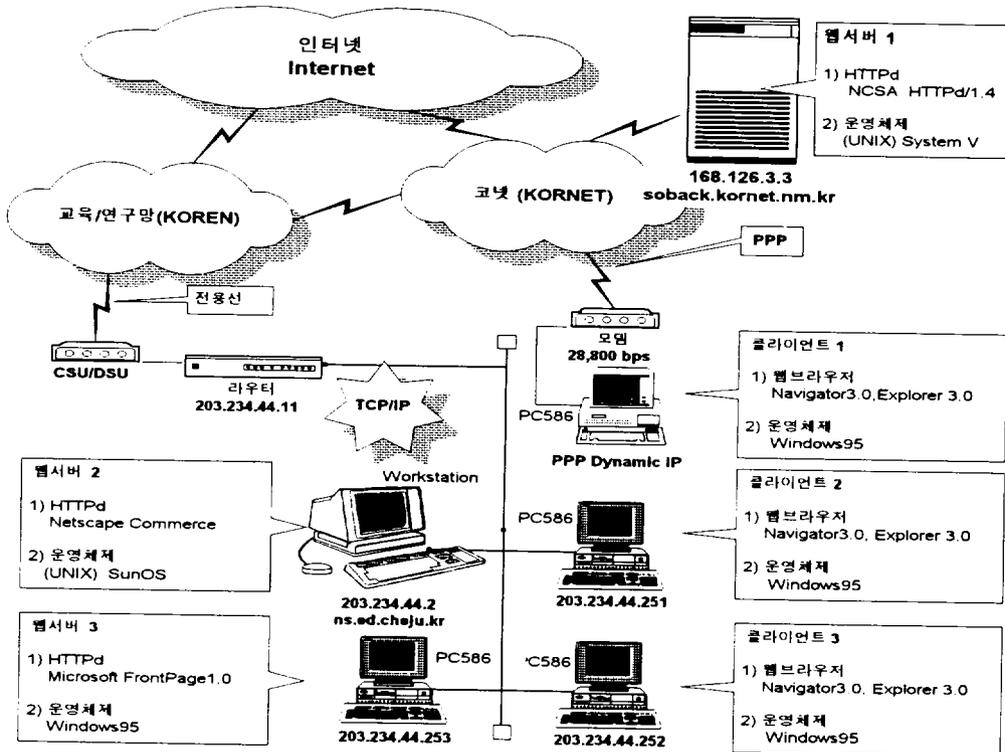


그림 2. 제작한 HTML문서를 탑재한 웹서버와 검색 클라이언트의 배치

② 접속 회선의 종류에 따른 결과

· 동일 LAN상의 서버에 접속

애플릿이 실행되는데 까지 10-20초의 시간이 걸렸다.

서버(203.234.44.2, 203.234.44.253)

클라이언트(203.234.44.252)

LAN(Ethernet 10Mbps)

· 가정에서의 전화선 접속

회선 여건이 최상일 때 약 40초의 시간이 걸렸다.

서버(203.234.44.2, 203.234.44.253)

클라이언트(KORNET의 제주PPP서버에 접속한 가정의 컴퓨터)

모뎀(28,800bps)

· 전용선을 통한 KORNET의 웹서버 접속

회선여건이 최상일 때 약 20초의 시간이 걸렸다.

서버(168.126.3.3)

클라이언트(203.234.44.252)

전용선(256kbps: 제주도교육청-제주대학교)

(2) '포사체 운동 실험기' 애플릿의 실행

일단 애플릿의 로딩이 완료되어 가동을 시작하면 서버, 네트워크, 클라이언트 등 제반 환경 조건에 관계없이 비슷한 실행 성능을 나타내었다. 클라이언트 컴퓨터의 기계적 성능에 따라 다소 차이가 나기는 하였으나 특기할 만한 정도는 아니었다. 그러나 Explorer3.0의 경우 몇가지 문제점이 발견되었다.

① 애니메이션

연산에 의한 포사체의 궤적이 원활하게 표현되었다.

② 화상 표현

고색상 표현 능력을 실험하기 위하여 16비트의 JPEG형식의 그림을 배경으로 사용하였다.

· Navigator 2.0 : 256색상으로 디더링(dithering)을 하였다.

· Navigator 3.0 : 그림의 본래 색상을 충실히 재현하였다.

· Explorer 3.0 : 그림의 본래 색상을 충실히 재현하였다.

그러나 대포나 표적 그림의 경우 최초의 애니메이션 시점에 이르러서야 비로소 파일을 읽어 오므로 학습자가 기다려야 하는 문제점이 발견되었다. mediaTracker를 사용하여 미리 자료를 읽어두도록 프로그램의 논리를 개선할 필요가 있다.

③ 음향재생 능력

.AU 형식의 8bit 11KHz 모노(mono)로 샘플링된 음향을 발사와 폭발음으로 사용한 결과 원활하게 재생되었다. 그러나 최초 음향 발생 시점에 이르러서야 음향 파일을 읽

어오므로 회선의 전송 속도가 느린 경우 학습의 흐름이 단절될 가능성이 발견되었다. 애플릿을 읽음과 동시에 음향 파일까지 미리 읽어 오도록 프로그램의 논리를 개선할 필요가 있다.

④ 조정판의 작동

애플릿이 안정화될 때에는 잘 작동하였으나, 불안정한 동안에는 학습자의 반응을 적시에 접수하지 못하는 경향이 있다. 특히 Explorer3.0의 경우 조정판의 한글이 나오지 않고 레이아웃(Layout)이 변해버리는 문제점이 발견되었다.

⑤ 애플릿 파일의 크기

그림과 음향 파일을 제외하였을 때 총 14kbyte(Fire.class:8kbyte,ControlPanel.class:6kbyte)의 크기를 차지하였다. 웹 환경이 아닌 시스템에서 이 정도의 성능을 나타내는 프로그램을 작성하려면 최소 100kbyte 이상의 실행파일 용량을 차지하게 된다. 네트워크에 관한 기능의 구현은 물론 윈도우 표현 및 조절 기능, 문자 및 화상 표현 기능, 음향 재생 기능 등을 포함시켜야 하기 때문이다. 웹 환경에서는 이러한 기능들이 이미 갖추어져 있을 뿐만 아니라, 자바를 지원하는 웹브라우저의 경우 자바 애플릿 실행 코드의 상당 부분이 웹브라우저 측에 이미 포함되어 있으므로 작은 용량의 애플릿으로도 고성능의 기능을 구현할 수 있게 된다. 실행 파일의 획기적인 축소는 컴퓨터 시스템 자원을 절약하고 통신 속도를 향상시키는 긍정적인 효과를 가져온다.

(3) '포사체 운동 실험기' 애플릿의 활용

① 애플릿의 가동

웹브라우저를 통하여 애플릿이 포함된 HTML 문서의 URL을 지정·호출하면 먼저 HTML 부분이 읽혀져 웹브라우저에 나타난다. 이어서 애플릿이 로딩된 후 가동되면서 HTML 문서의 출력 결과 위에 그림3)와 같은 조정판이 나타난다.

② 배경판으로 스크롤

포사체 운동 애니메이션

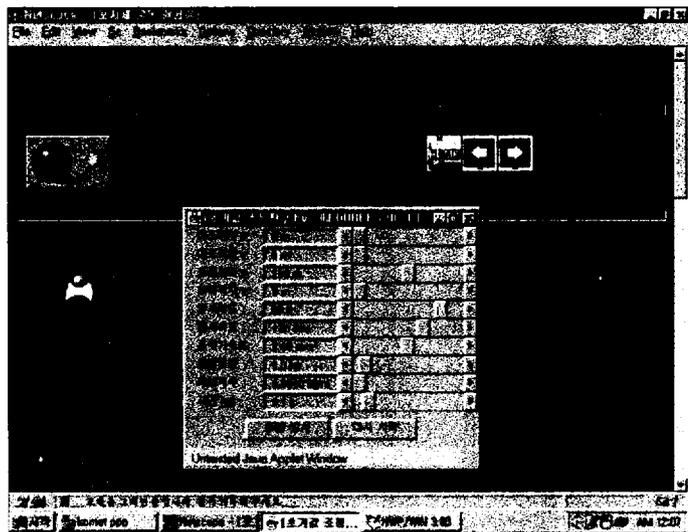


그림 3. '포사체 운동 실험기' 애플릿이 가동된 화면

선이 일어나는 배경판은 HTML 문서의 아랫 부분에 두었으므로 스크롤해야 볼 수 있다. 웹브라우저의 수직 스크롤바(scrollbar)를 누르면 다음과 같이 배경판이 나타난다. 이 때 조정판이 브라우저의 뒤로 숨어 버리므로 윈도우 하단의 조정판 아이콘을 클릭하여 그림 4)처럼 조정판이 웹브라우저의 위로 나타나게 해야 한다.

③ 조정판 조작

학습자는 조정판의 변수들을 설정해야 한다. 조정판에서 숫자 입력은 허용하지 않으며 해당 변수의 스크롤바를 마우스로 좌우 방향으로 끌어서 값을 변화시켜야 한다. 마우스로 스크롤바를 끌어도 잘 작동하지 않을 때에는 애플릿이 안정될 때까지 기다려야 한다.

④ 포탄의 발사

발사버튼을 마우스로 선택하면 발사 음향과 함께 포사체 운동 궤적이 그려지기 시작한다. 포탄의 발사 회수에 따라 궤적의 색깔이 다르므로 실험 결과를 비교하는 것이 쉽다. 포사체 운동 궤적은 점으로 이루어 지는데, 조정판의 '시간(dt)'의 값을 변화 시킴으로써 점이 찍히는 간격을 바꿀 수 있다. 시간 간격을 작게 할수록 정밀

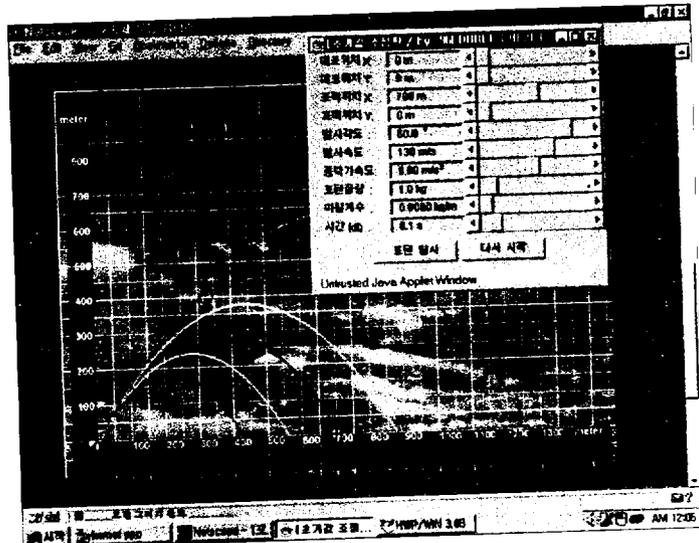


그림 4. 조정판과 포사체 운동 애니메이션

한 결과를 얻을 수 있으나, 순간의 속도 변화 등을 비교하려는 목적이라면 시간 간격을 너무 작게하는 것은 좋지 않다. 시간 간격을 작게하면 점 사이의 간격을 측정하기가 곤란해지기 때문이다. 그리고 두 개 이상의 궤적을 시간에 따라 비교하고자 할 때에도 어느 정도 시간 간격을 두어 점들이 뚜렷하게 구분되도록 하는 것이 좋다. 일반적인 실험 목적이라면 0.1초에서 0.3초 사이의 값이 적당하다. 기존의 궤적을 모두 지우고 새롭게 시작하고자 할 때에는 리셋(reset)버튼을 선택하면 된다.

⑤ 실험기 설명서

보통 이러한 실험기 주변에는 실험기를 자세히 설명하는 도움말을 포함시켜 주는 것

이 좋다. 가급적 HTML 문서의 형태를 취하는 것이 좋는데 내용의 변경과 확장이 매우 자유롭고 그림이나 애플릿과 같은 요소를 삽입하여 내용을 풍부하게 할 수 있으므로 바람직하다. 그림5)은 '포사체 운동 실험기'의 사용법을 설명하는 HTML문서로 그에 대한 링크를 실험기 애플릿의 옆에 배치하였다.

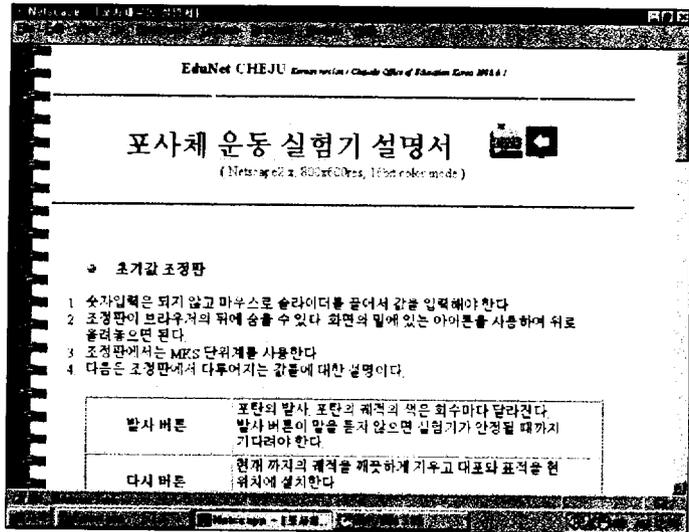


그림 5. '포사체 운동 실험기' 설명서 (HTML 형식)

3) '포사체 운동 실험기' 애플릿의 활용 사례

포사체 운동은 2차원 운동의 예로서 물리 과목의 서두에 전개되는 경우가 많다. 포사체 운동에 관한 학습에서는 보통 힘의 방정식으로 부터 출발하여 과제를 해결하게 된다. 실험 활동이 병행되면 훨씬 효과적임에도 불구하고 포사체 운동을 직접 실험하는 것이 곤란하여 방정식의 해를 구하는 선에서 만족해야 한다. 많은 학생들은 이러한 추상성을 납득하지 못하고 물리 과목을 경원시하게 되는데 포사체 운동의 시뮬레이션은 이러한 경우 매우 유용하다.

(1) 발사각도의 변화 실험

발사 각도에 따른 사거리(射距離)의 변화와 최고 도달 높이에 관한 관계는 포사체 운동에서 흔히 다루

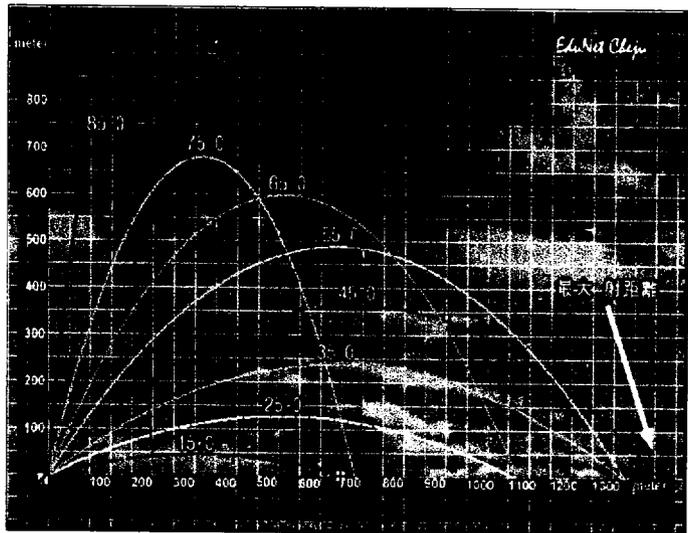


그림 6. 발사 각도를 변화시키면서 실험한 결과 화면

어지는 과제이다. 학생들은 '포사체 운동 실험기' 애플릿의 조정판을 이용하여 대포의 발사 각도를 여러 가지로 바꾸면서 실험을 하고, 그 결과 나타나는 궤적에서 원하는 측정치를 얻어 낼 수 있다. 그림6)은 발사 속도를 120m/s로 고정시키고 발사 각도만 변화시키면서 얻은 포탄의 궤적을 보여 준다. 한 눈금의 크기는 50m이다. 궤적 위의 각도값은 실행 화면을 갈무리한 후 그래픽 편집기로 써 넣은 것이다. 수식에서 유도되는 바와 같이 45도에서 최대 사거리를 나타낼 수 있는데 흥미있는 것은 55도와 35도, 65도와 25도, 75도와 15도가 각각 동일한 사거리를 나타내면서 대칭성을 보인다는 점이다.

(2) 중력가속도에 따른 포사체 운동의 차이

시뮬레이션의 유익한 점은 중력과 같이 실험실에서는 조작할 수 없는 변수를 여러 가지로 바꾸어 볼 수 있다는 점이다. 그림 7)에서는 그림 위에 표시한 바와 같이 중력가속도의 값을 0m/s²에서 18.80m/s²까지 변화시키면서 발사 실험을 수행한 결과를 나타내 보이고 있다. 여러 중력값에 의한 포사체 운동의 궤적에서 포물선 운동

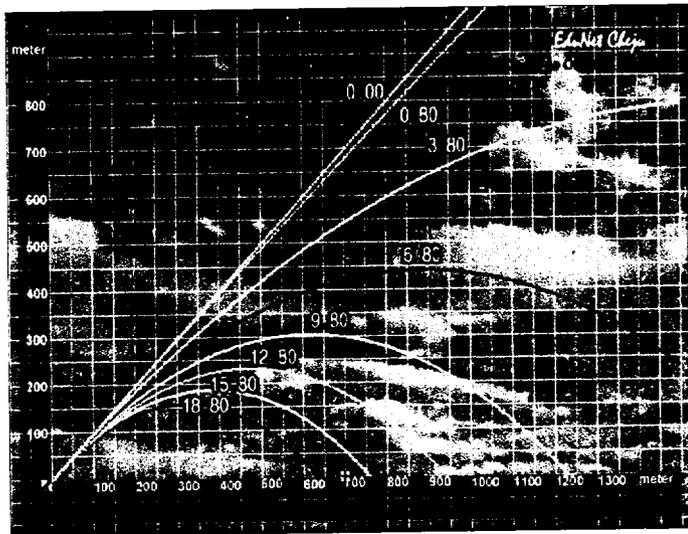


그림 7. 중력가속도를 변화시키면서 실험한 결과 화면

이 중력 때문에 나타난다는 것을 뚜렷하게 보여주고 있는데, 이는 방정식의 해를 구하는 과정보다 학생들에게 더욱 구체적으로 중력의 효과를 이해시킬 수 있음을 뜻한다. 특히 무중력 상태를 뜻하는 중력가속도 0m/s²에서의 포사체의 운동은 다른 궤적과 비교되어 인상적이며 힘을 받지 않는 물체는 등속운동을 한다는 사실을 극적으로 보여주고 있다.

(3) 운동의 합성

포사체의 운동을 수평 방향의 등속 직선운동과 수직 방향의 자유 낙하운동의 합성 운동으로 해석해보는 것은 여러 가지 면에서 유익하다. 그리고 이에 관련된 학습에서는 자유 낙하운동과 포사체 운동이 동시에 나타나는 섬광 사진을 이용하는 경우가 많다. 본 연구에서 제작한 '포사체 운동 실험기'는 중력가속도와 발사 속도는 물론 점이 찍히는 시간

간격을 조절할 수 있기 때문에 이러한 용도로 활용 가능하다. 그림8)은 발사속도를 100m/s, 발사각을 0도로 고정하고 점이 찍히는 시간 간격을 0.5초로 하여 중력가속도를 달리하면서 실험한 결과이다. 그림에서 보는바와 같이 중력에 관계 없이 수평 방향의 100m 당 점의 개수는 2개이다. 이는 모든 경우 수평 방향으로의 등속운동을 하고 있음을 나타낸다.

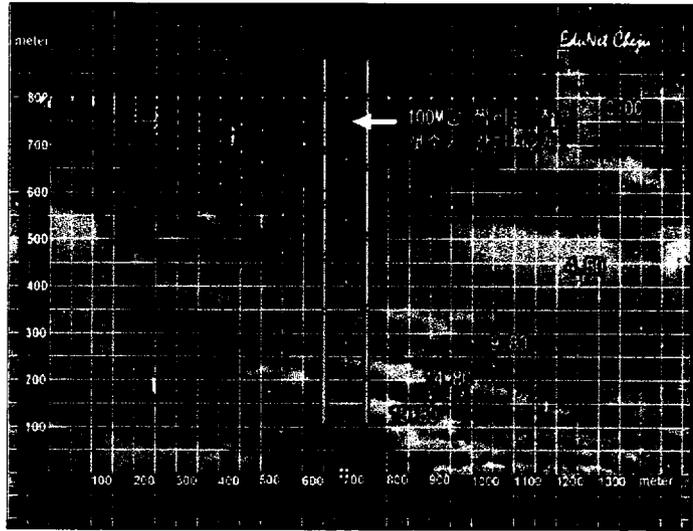


그림 8. 시간간격을 크게 잡아 섬광사진 효과를 얻은 실험 화면

포사체 운동과 자유 낙하운동 그리고 등속 직선운동을 동시에 비교하고 싶으면 다음과 같이 값을 설정하여 실험하면 된다.

- 자유 낙하운동 - 발사각 0도, 발사 속도 0m/s, 중력가속도 $9.8m/s^2$
- 등속 직선운동 - 발사각 0도, 발사 속도 100m/s, 중력가속도 0m/s²
- 포사체 운동 - 발사각 0도, 발사 속도 100m/s, 중력가속도 $9.8m/s^2$

그림9)은 시간 간격을 0.5초로 하여 세가지 운동을 비교한 실험 결과이다. 그림 상에 표시된 바와 같이 8초 후 포탄의 수평 좌표는 등속 직선운동을 따르고 수직 좌표는 자유 낙하운동을 따르는 것으로 나타나 있다. 단, 후에 발견된 애플릿 프로그램의 오류로 인하여 자

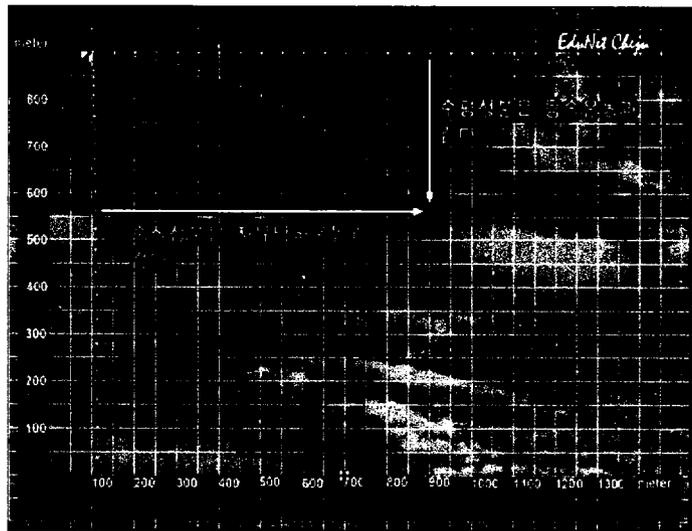


그림 9. 등속운동, 낙하운동과 포사체운동을 비교한 실험

유 낙하운동의 경우 수평속도를 1m/s로 하였다.

(4) 공기의 저항력이 있는 경우의 포사체 운동

본 실험기는 매 순간마다 속력을 계산하여 위치를 정하는 방법을 택하였으므로 저항력의 영향도 도입하는데 간단한 프로그램으로 해결할 수 있었다. 그러나 저항력의 크기는 물체의 속도, 모양과 표면 상태, 물체의 질량, 공기의 상태 등 여러 가지 요인의 영향을 받아서 그 양상이 매우 복잡하므로 정밀성은 크게 고려하지 않았다. 저항계수는 포탄 속도의 제곱에 비례하는 것으로 정하였다. 그림10)는 포탄의 발사 속도와 발사각, 중력가속도를 고정하고 저항수만 일정하게 증가시키면서 실험한 결과이다. 저항계수의 값이 커짐에 따라 운동에의 저항력은 급격히 커져서 사거리가 짧아짐과 동시에 포사체의 궤적은 비대칭적으로 변하는 것이 잘 드러나고 있다.

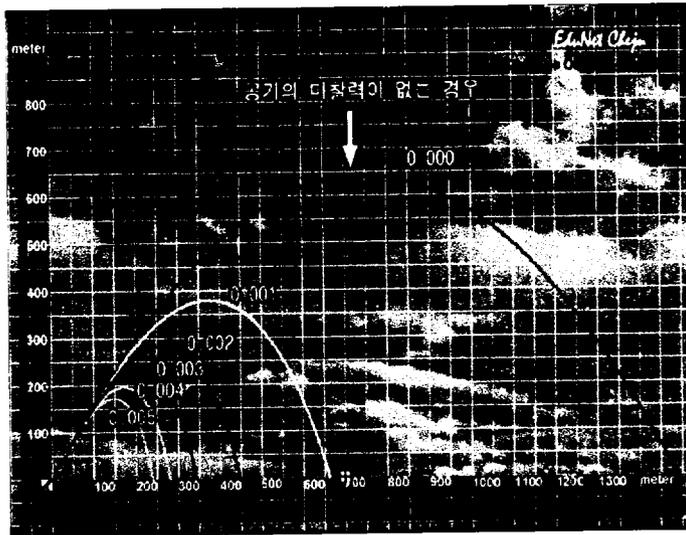


그림 10. 공기 저항의 증가와 포사체 운동의 변화 실험

그림10)는 포탄의 발사 속도와 발사각, 중력가속도를 고정하고 저항수만 일정하게 증가시키면서 실험한 결과이다. 저항계수의 값이 커짐에 따라 운동에의 저항력은 급격히 커져서 사거리가 짧아짐과 동시에 포사체의 궤적은 비대칭적으로 변하는 것이 잘 드러나고 있다.

V. 결론

중등학교 물리교육 CAI 코스웨어를 개발하는데 활용할 수 있는 웹의 자원과 사례를 분석하였다. 기존의 웹 환경에서는 통신, 다양한 자료의 출력, 분기, 상호작용에 관한 기능이 기본적으로 제공되므로 적극적으로 활용할 필요가 있으며 다른 종류의 인터넷 서비스와 확장 기능들을 적절히 조합·활용함으로써 코스웨어의 질은 물론 개발의 생산성을 높이는 것이 바람직한 것으로 나타났다. 그리고 물리 시뮬레이션과 같이 역동적인 애니메이션과 강한 상호작용 및 연산처리를 구현하기 위한 프로그램 작성에서는 클라이언트측과 서버측에 프로세서를 적당히 안배하는 것이 중요하며 자바와 같은 분산처리지향의 언어를

도입하는 것이 바람직한 것으로 나타났다.

시뮬레이션형 물리교육 CAI코스웨어를 웹에서 구현하기 위해서 수반되는 기술적 문제점을 파악하기 위하여 예시적인 자바 애플릿인 '포사체 운동 실험기'를 제작하였다. 이 실험기는 16비트 색상의 화상과 샘플링된 음향은 물론 시뮬레이션에 필요한 애니메이션을 원활하게 표현할 뿐만 아니라 파일 크기가 14kbyte로 매우 작아 서버로부터 클라이언트로 전송되어 가동되는데 불과 1분 이내의 시간이 소요되었다. 학습자가 여러 가지 물리 변수를 설정할 수 있는 상호작용 기능이 제공되며 1% 이내의 실험 오차를 보이므로 정량적인 모의실험 활동에 활용할 수 있을 것으로 판단되었다. 특히 점이 찍히는 시간 간격을 충분히 주어 포사체의 시간에 따른 위치가 구분될 수 있도록 하면 섬광 사진에 의해 포사체 운동을 분석하는 효과를 얻을 수 있으며 이전 궤적이 지워지지 않기 때문에 다른 조건에서의 실험 결과를 서로 비교하는 것이 용이하였다. 이 실험기를 사용하면 발사속도와 각도에 따른 포사체 운동의 변화에 대한 학습은 물론, 중력가속도의 값을 바꿀 수 있으므로 힘이 운동에 미치는 영향을 학습하는데 도움이 될 것으로 보인다. 더구나 포사체 운동을 x,y축에 따른 운동으로 분석해 보는 등 응용 방법이 다양하므로 '포사체 운동 실험기' 애플릿은 월드와이드웹용 물리교육 CAI 코스웨어를 제작하는데 그 활용 가치가 크다고 판단된다.

참고 문헌

1. Nand Lal, Digital Library Technology, <http://dlt.gsfc.nasa.gov/> (1996).
2. Fritz Hasler, About RSD Program, <http://rsd.gsfc.nasa.gov/rsd/About.html> (1996).
3. 김평철, WWW 96-1, 3, 웹코리아(WWW-KR), p.50 (1996).
4. B.Ibrahim, Pedagogical Value of the World-Wide Web, http://cui_www.unige.ch/eao/www/Bertrand.html (1994).
5. 이택경, WWW 96-1, 3, 웹코리아(WWW-KR) pp.35-49 (1996).
6. J.Newmarch, Client-Side Execution of Programs, <http://pandonia.canberra.edu.au:80/ClientSide> (1995).
7. T.Martin, The Development of Interactive World Wide Web Courseware for Students of Engineering and Technology at Deakin University (1995).
8. B.Ibrahim, Advanced Educational Uses of the World-Wide Web, http://cui_www.unige.ch/eao/www/Bertrand.html (1994).
9. D.Wojtowidz, R.Wilhelmson, M.Ramamurthy, IICE: Bring Interactivity of Image-Based WWW Products (1995).
10. J.Gosling, H.McGilton, The Java Language(tm): White Paper, <http://www.javasun.com/doc/java-whitepaper-1.html> (1995).