

Mathematica를 이용한 웹기반 미적분 모듈의 개발

전영국[†]

요 약

본 논문에서는 Mathematica를 자바 언어로 연결하는 J/Link를 이용하여 웹기반 환경에서 미적분 문제의 해를 단계별로 제시하는 모듈을 소개하였다. 이 모듈은 중하위 수준의 학생들에게 미분 문제의 해를 단계별로 제시함으로써 문제해결의 과정을 단계별로 연습할 수 있는 환경을 제공하고 있다. 또한, 학생들에게 미분 연산자를 적용할 함수와 미분된 결과 함수에 대한 그래픽 결과를 보여줌으로써 미적분 개념에 대한 이해의 폭을 넓히는 효과를 노렸다. 수학 전문가시스템인 Mathematica는 미분 문제의 해결 과정을 단계별로 제시함으로써 수학 지식을 체계적으로 제공하는 기반을 마련하고 있으며, 수학 학습용 튜토리얼 또는 CMI로 연계될 수 있는 가능성을 보여주고 있다. 또한 이 시스템은 FrontEnd에서 처리되는 수식 폰트를 자바서브릿에서 처리함으로써 웹환경에서 수식 기호를 우회적으로 구현하는 방법을 제시하고 있다.

Development of a Web-based Calculus module using Mathematica

Youngcook Jun[†]

ABSTRACT

This paper illustrates a calculus module which generates step-by-step solutions using J/Link that connects Java and Mathematica. Such a module provides intermediate and low level students with a practical environment where they can easily follow the solution paths on their own paces. The extra feature of this module depicts graphical images for a given function and its differentiated result to enhance the visual understandings of calculus concepts. Mathematica as a mathematical expert system that provides systematic mathematical knowledge to students with step-by-step solutions will be possibly extended to the tutorial or CMI development. The proposed module is implemented in a Java servlet that links to Mathematica FrontEnd. This approach results in adopting font systems to express two dimensional mathematical expressions in web documents as an alternative typesetting tool.

1. 서 론

컴퓨터대수 시스템(Computer Algebra System, 이하 CAS)이란 컴퓨터를 이용하여 수학에 관련된 문제를 기호로 처리하여 연산을 수행하는 컴퓨터 시스템으로서 수치계산과 대수적 연산을 수행하고 필요한 경우에 연산의 결과를 그래픽으로 처

[†] 종신회원: 순천대학교 컴퓨터교육과 조교수
논문접수: 2001년 6월 25일, 심사완료: 2001년 8월 26일

리하는 소프트웨어이다. 이러한 컴퓨터대수 시스템에 속하는 소프트웨어로서 대표적인 것이 Mathematica, Maple, Reduce, Derive, Maxima 등이 있다. 컴퓨터대수 시스템을 이용하여 수학교육에 적용하는 것에 대한 연구는 그다지 흔하지 않다. 컴퓨터대수 시스템의 개발과 별도로 수학교육에 적용된 소프트웨어는 여러 가지 종류가 있으며, 이러한 소프트웨어는 자체적으로 수식표현과 연산 처리 및 그래픽을 위한 기능을 따로 구현하는 것이 일반적이다. 그러나 컴퓨터대수 시스템은 주로 교육용보다는 연구용으로 제작된 특징이 있기 때문에 수학교육에 접목되는 부분은 많은 연구를 필요로 한다(Wester, 1999).

대표적인 CAS인 Mathematica와 Maple를 간단히 비교하면 다음과 같다(Heck, 1993; Wolfram, 1991). 이 두 시스템의 공통된 특징은 기호연산과 수치연산을 위한 커널이 있고, 수식 입력과 출력을 위한 인터페이스가 사용하기 쉽게 제작되었으며, 수식 폰트와 그래픽 처리가 뛰어난 장점을 지니고 있다. 사용자가 입력한 수식을 인터프리터 형태로 파싱하여 결과를 처리하는 점과 자체적으로 가지고 있는 프로그래밍 언어와 다른 포맷으로 문서를 변환하는 기능을 갖추고 있어서 확장성과 호환성의 특징이 돋보인다. Mathematica는 Mathlink라는 API를 이용하여 Mathematica의 내장된 명령어를 다른 프로그램에서 호출하여 실행할 수 있으며, Maple 6.0 버전은 Excel 2000에 한정하여 서로 명령어를 호출할 수 있는 기능을 내장하고 있다.

Mathematica의 기능을 살려서 출력 결과를 웹 브라우저에 전송하는 방식을 토대로 하여 CAI 프로그램을 설계하고 구현한 내용은 전영국의 논문(1997)에서 보고되고 있다. Mathlink를 이용하여 Mathematica의 기능을 웹환경으로 확장한 이유는 현재 웹전용 Mathematica 버전이 나와 있지 않으므로, 웹환경에서의 수학교육용 자료 개발의 설계 및 구현을 시도하는 것이 중요하기 때문이었다. 그리고 Mathematica에 내장된 뛰어난 그래픽 기능을 사용하여 수학 개념의 시각화를 촉진하는 요소도 고려되었다.

본 논문에서는 Mathlink의 자바버전인 J/Link를 이용하여 Mathematica의 FrontEnd 인터페이스에서 제공하는 수식폰트를 웹상에서 출력하고, 미분문제를 단계별로 해결하는 패턴매칭의 패키지를 구현한 방법을 소개하고 있다. 사용자가 입력하는 다양한 형태의 미분문제를 그래픽과 단계별 수식으로 제공하고 각 단계에 해당되는 규칙을 보여줌으로써 일종의 설명을 제공하는 효과를 내고 있다. 이런 방법은 초보자와 중간 단계의 학습자에 많은 도움을 줄 수 있으리라는 가설로 출발하고 있다.

2. 기존 연구

2.1 수학교육에서의 CAS 활용

일반적으로 컴퓨터대수 소프트웨어를 이용하여 단계별 계산을 하는 것이 필요한 것인가 그렇지 않은가에 대한 논의는 이전부터 있어왔다. 수학 문제를 해결하는 알고리즘을 이해한 학생이라면 수식의 결과만 신속하게 처리하는 것이 다른 수학의 개념을 탐구하는데 더 도움이 될 것이지만, 문제해결의 단계를 이해하지 못하는 학생은 손으로 직접 문제를 풀어보면서 단계별 풀이과정을 습득하는 것이 바람직하다. 기호연산을 조작함으로써 풍부한 수학적 표현을 경험하게 되고, 대수적 기호 처리를 기하학적 표현으로 변환하는 과정을 살펴봄으로써 수학개념의 이해를 증진시키는 것이 컴퓨터대수 시스템이 수학교육에 도입되는 근거가 된다. 수학적 표현에 가까운 프로그래밍 언어를 도입하여 대학수학의 변화를 시도할 목적으로 개발된 것이 ISETL이다(Dubinsky, 1995).

적분과 미분의 개념을 이해하고 각 단계의 알고리즘을 이해하고 있다면 굳이 중간 단계의 계산을 일일이 하지 않아도 되지만, 단계별 수식의 전개가 이해되지 않는 학생은 수식변환의 과정을 차근차근 살펴보는 것이 학습에 도움이 될 것이다(Buchberger, 1989). 이와 같은 단계별 풀이 과정을 규칙을 이용한 패턴 매칭으로 처리하는 것은 rewriting system에 속한다. Linear Kid(Jun,

1995)도 그와 같은 시스템에 속하는데, 주로 중·하위권 학생에게 단계별 풀이 과정이 학습에 도움이 되는 것으로 보고되었다. 단계별로 해를 구하여 나가는 방식은 주어진 식으로부터 간단한 수식으로 변환하는 것으로 정의할 수 있는데, 이것은 단순화(simplification)이라는 용어로 정의하여 수학적으로 구명할 수 있다(Buchberger, & Loos, 1982).

한편 컴퓨터대수 시스템이 수학교육에 적용되는 연구에서 몇 가지 제한점 및 단점들이 지적되고 있다. 기호 처리를 하는 과정에서 대수식 표현을 대수식에 대응되는 그래프로 표현하는 시각화 부분은 원래 대수식의 다른 변형에 불과하다는 지적이 있다. 수학적 개념이 과정과 대상을 동시에 함축한다는 "procept"를 주장한 Tall(1991)에 의하면 대수식을 단순히 그래프로 보여주는 것은 수학학습에 결정적인 촉진요소가 되지 않고, procept를 시각적으로 보여주는 그래픽이 학습에 도움이 된다고 주장한다. 예를 들면, 매개변수 t 로 표현된 3차원 곡면의 한 점이 있을 때 이 점에 접하는 평면에서의 differential($dy/dx = (dy/dt)/(dt/dx)$)을 시각적으로 표현하는 것이 기호로 표현하는 것보다 접평면에 관련된 변화율에 관한 이해를 증진시킨다.

네덜란드의 RME(Realistic Mathematics Education)²⁾에서 적용한 그래픽 계산기의 활용에 대한 보고서에 의하면, 그래픽 계산기³⁾에 탑재된

2) RME(Realistic Mathematics Education)라고 불리는 이 방법은 네덜란드에서 수평적, 수직적 수산화 과정을 교수 가능한 형태로 이용하는 현실적 수학교육 프로그램으로 개발되었다(Treffers, 1991). 개념의 발달은 비형식적, 서술적 상황(contexts)에 관련된 경험으로부터 수학적 형식주의로 진행해 간다. 주어진 상황에 관련된 문제를 다루는 학생들은 그들의 경험으로부터 하나의 모델을 수평적으로 수산화하는 과정을 겪게 된다. 수직적 수축화를 통해서 이 모델은 변환되고 또한 학생들에 의해 형식적인 수학적 구조로 변환된다. 현실적 수학교육(RME)의 강조점은 학생의 현존하는 활동방식들과 수학적 사고를 연결하고 자연스럽게우면서 더욱 일반화와 추상화를 촉진하는 서술적인 상황들을 사용한다는 것이다. 이 RME에 CAS가 탑재된 그래픽 계산기를 사용하는 것이 중요한 이유 중의 하나이다.

3) 참고로 TI, HP, Casio와 같은 그래픽 계산기에는 CAS가 탑재된 모델이 있다. 그 중에서 TI-92에는

컴퓨터대수 시스템을 사용하는 학생들이 다음과 같은 어려움을 겪고 있음이 나타나고 있다(Drijvers, 1999). 첫째, 교과서에서 나타나는 수식 표현과 CAS에서 보여주는 수식 표현 사이에 차이점이 있으며, CAS가 구현된 과정에서 대수적 연산과 수치적 연산의 차이점이 암시적으로 나타나고 있다. 둘째, 대수적 기호 처리를 하는 학생에게 적절한 전략을 제공하는데 한계가 있으며, 언제 어떻게 CAS를 사용하는 것이 바람직한가에 대한 안내를 하는 것이 어렵다.

2.2 웹기반 폰트 처리

다른 컴퓨터대수 시스템과 마찬가지로 Mathematica는 수학에 사용되는 서체(font)를 TEX 및 LaTeX 등으로 다양하게 표현하는 기능을 가지고 있다. 수식 팔레트를 이용하여 입력하는 방식을 사용하면 사용자는 한층 쉽게 수식을 표현하게 되고, 그 결과를 Notebook에 출력하게 된다. 웹문서(html) 변환은 "File" 메뉴 항목중 "Save As Special..."을 선택하고 다시 "HTML" 항목을 선택하면 변환이 가능하다. Maple도 이 기능을 가지고 있으며, 특히 애니메이션 부분이 그대로 gif 파일로 변환되는 점이 돋보인다. 3차원 그래픽을 VRML2.0으로 변환하는 기능이 Maple에는 내장되어 있으나 Mathematica는 별도의 패키지를 사용하여야 한다. 한편 Mathematica의 3차원 그래픽을 자바로 구현된 Live3D로 변환하면 VRML과 같은 효과를 거둘 수 있다(<http://mathworld.wolfram.com/notebooks/LiveGraphics3D.m>).

현재 웹문서 작업에 필요한 수식편집의 방식을 소개하면 다음과 같다. 혼글에서 수식편집기를 사용하여 작성한 파일을 나모웹에디터로 웹문서로 변환하는 방법이 쓰이고 있다. MS Office 계열의 소프트웨어에서 수식편집기를 사용한 다음, 워드를 사용하여 웹문서로 변환하는 방법도 가능하다. 이 경우에 각각 수식편집기를 장착하고 있

Lisp으로 구현된 Derive가 있으며, 기하 소프트웨어로는 Cabri Geometry 또는 GSP가 포함되어 있다. 중등학교의 수학수업에 그래픽 계산기를 활용에 대한 부분은 황우형(1997)을 참조하기 바란다.

다. 일반적인 저작도구에도 자체의 수식편집기를 가지고 있는 경우가 있는데, 멀티미디어 저작도구인 PASS2000이 그런 경우에 속한다. 이 파일은 PASS2000을 구동시키는 플러그인을 설치한 웹브라우저에서 실행이 가능해진다. 웹에서 수식을 표현하는 가장 간단한 방법은 수식을 gif 파일과 같이 그래픽으로 처리한 다음 웹문서에 직접 연결시키는 inline image 기법이다. 이 방법은 다양한 수식표현을 일일이 수작업으로 처리해야 하는 불편함과 비효율성이 문제가 된다. 그 외에도 자바애플릿을 이용한 WebEQ, IBM techexplorer를 이용하는 방법 등이 제한적으로 도입되고 있다.

2.3 웹 수식표현의 동향

수식표현을 그래픽으로 처리하면 간단히 그 결과를 웹문서에 표현할 수 있으나, 동적으로 변화하는 수식을 출력하는 경우에는 매우 복잡한 문제를 야기시킨다. 특히, 그래픽으로 처리된 수식의 폰트 크기, 스타일 등을 다양하게 표현하기 어렵고, 어떤 경우에는 수식을 알아보기 어려울 뿐만 아니라, 수식의 내용을 다른 컴퓨터대수 시스템을 이용하여 처리하기가 불가능해진다. 이런 어려움을 해소하기 위하여 현재 표준화 작업이 진행되고 있는 영역이 OpenMath와 MathML이다. 이 표준화 작업은 컴퓨터 플랫폼에 종속되지 않고 모든 사람이 쓸 수 있도록 무료로 공개되어야 하며, 수학적 의미를 명확히 담겨 있어야 하고, 확장가능성이 있어야 한다. 또한 LaTeX와 같이 다른 수식처리에 호환이 되어야 하며, 수식 처리가 적절한 수준에서 효율적이어야 한다(Le, 1999).

XML(Extensible Markup Language)에 기초를 둔 OpenMath(<http://www.openmath.org/>)와 MathML(<http://w3c.org/Math>)은 위에서 제시한 표준 지침에 따르는 공통점이 있다. $(a + b)^2$ 의 예를 보면 수식을 왼쪽에서 오른쪽으로 파싱하여 제시하는 방법과 “a+b”에 대한 거듭제곱이라는 의미론적 표현의 방법이 모두 고려되는 것이 바

람직하다. 이 부분 역시 OpenMath와 MathML의 공통점이 된다. 두 방법의 차이는 기술적인 구현상의 차이에 기인하는데 수식표현에 따른 속성(폰트 크기, 스타일, 색깔) 위주로 구현하느냐 또는 수식의 유형(type) 정보에 따라 의미론적 구현(예, DTD를 확장한 Content Dictionary)에 치중하느냐 등에 따라 달라진다. 그러나 어느 방법 이든지 가능한 모든 수식 표현을 구현해 내는 것이 무엇보다 중요하므로, 오랜 시간이 걸리더라도 이 방식의 연구가 지속될 것임에는 틀림없다.

3. 모듈 설계 및 구현

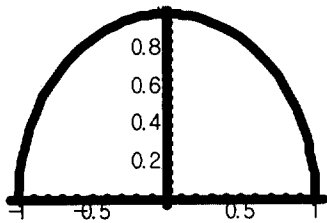
본 연구에서 설계한 모듈은 크게 세 가지의 특징을 가지고 있다. 첫째, 미적분의 문제를 해결하는데 함수의 그래프를 같이 제공함으로써 시각화의 기능을 추가한 점이다. 둘째, 미분 문제를 하는 과정을 간단한 주석과 함께 단계별로 제시함으로써 학생들이 단순한 계산보다 미분의 과정에 치중할 수 있도록 하였다. 이것은 Linear Kid의 연구에서 나타난 바와 같이 중하위권 학생들을 주대상으로 모듈을 설계하였다(전영국, 1995). 셋째, 웹상에서 2차원 수식표현을 구현하는 수학문서를 작성하는 것은 J/Link를 사용하여 Mathematica의 인터페이스를 자바서브릿에서 처리하도록 설계하였다.

3.1 Mathlink를 이용한 적분의 모듈

적분 문제를 입력하면 적분의 결과를 제시하는 아래의 웹사이트를 사용하면 그래픽 결과와 함께 볼 수 있다. 예를 들면, (그림 1)처럼 반원을 나타내는 $\text{Sqrt}[1 - x^2]$ 를 적분해보자. “Integrate[Sqrt[1 - x^2], x]”를 입력하면 적분의 결과는 $(1*x*\text{Sqrt}[1 - x^2])/2 + \text{ArcSin}[x]/2$ 의 꼴로 나타난다. 이 사이트는 (그림 2)와 같이 주어진 함수와 적분한 결과에 대한 그래프를 동시에 제공하여 시각적 비교를 손쉽게 할 수 있도록 구성되었다. 단순히 그래프를 제시하는 것은 Tall(1991)의 지적대로 학습에 그리 도움이 되지 않으므로, 적분의 개념을 Sum의 극한 개념으로

바꾸어 단계별로 시각화하는 방안이 고려되고 있다(송기홍, 1995).

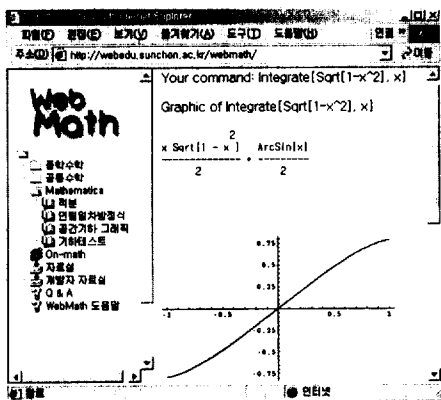
여기서 사용자가 적분할 대상을 웹인터페이스에 입력하면 CGI 프로그램이 그 입력을 Mathematica의 커널에 전달하여 그 결과를 받아서 웹문서에 출력한다. 이 과정에서 CGI 프로그램과 Mathematica의 커널 사이에 Mathlink 프로토콜로 연결되어 실행된다 (전영국, 1997). Mathlink를 사용한 이 방법과 달리 적분의 결과를 다양한 포맷으로 출력하는 사이트가 <http://www.integrator.com>이다. 이 사이트는 다음에서 설명할 J/Link와 자바서버릿을 함께 사용하여 수식의 출력을 stream 방식으로 처리하고 있다.



(그림 1) Plot[Sqrt[1 - x^2], {x, -1, 1}]의 결과

단한 다른 형태로 바꾸는 Mathematica 명령어는 Simplify인데, 이 명령어는 다른 컴퓨터대수 시스템에도 필히 구현되어 있는 중요한 명령어이다. 이 명령어는 Mathematica와 Maple에서 각기 다르게 설계되어 있는데 그 이유는 수식표현의 단순화 정도가 어디까지로 한정할 것인지에 대한 명확한 기준이 없기 때문이다. 이것은 사용자 입장에서 단순한 수식표현이 다항식 형태가 될 수도 있고, 인수분해 형태로 인수의 곱으로 나타나는 것을 원할 수도 있기 때문이다. 다음의 예를 보면 $1/(x*\text{Sqrt}[c*x^2 + b*x]^3)$ 을 적분하고 다시 미분한 결과를 간단하게 해보면 그 결과는 원래 함수와 같지만 표현이 약간 다르게 나타남을 볼 수 있다.

$$\frac{1}{x \sqrt{cx^2 + bx}^3} = \frac{1}{x^2 \sqrt{cx^2 + bx}}$$



(그림 2) 반원에 대한 부정적분 처리 모듈

적분과 미분의 기호연산에서 주의할 것은 기호연산의 중간 단계에서 발생하는 수식의 표현이 동치(equivalent)임에도 불구하고 똑같이(same) 나타나지 않을 수 있는 점이다. 어떤 수식이 간

3.2 J/Link를 이용한 단계별 미분 문제의 해결

적분은 Mathlink를 사용하여 구현하였으나, 단계별 미분 문제의 해결은 J/Link를 이용하여 자바 서버릿으로 구현하였다. J/Link는 Mathlink의 자바 버전으로 JLink.jar 파일과 JLink.m 패키지 및 JLinkNativeLibrary.dll 라이브러리로 구성되어 있다. 단계별 미분 문제의 해결은 Linear Kid와 같이 단계별로 적용되는 규칙을 제시하고, 교과서에서 나타나는 수식표현이 웹상에서 구현되고 있다(전영국, 1995). 예를 들면, “ $2x^2-1$ ”을 x에 대하여 미분하는 과정을 단계별로 제시하여 그 결과가 $4x$ 가 되는 과정을 자세하게 보여준다. 각 과정에서는 미분이 적용된 규칙을 첨부하여 단계별로 이행하는 설명을 대신하도록 하였다.

조금 복잡한 형태의 $\text{ArcTan}[2x^2-1/(2x^2+1)]$ 은 (그림 3)에서 보는 바와 같이 여러 단계에 걸쳐

서 미분이 전개되고 있다. 이와 같이, 매우 긴 과정의 미분 결과를 일일이 자세히 보여줌으로써 계산 과정에서 범하기 쉬운 오류를 정확하게 확인할 수 있고, 계산에 소요되는 시간을 다른 수학 개념의 습득에 활용할 수 있는 것이 이 모듈의 장점이 된다. 또한 교사는 일일이 손으로 계산하면서 과정의 풀이를 제시해야 하는 번거러움을 줄일 수 있고, 학생은 계산의 결과를 다시 열람하여 복습할 수 있는 노트 기록으로 활용할 수 있다.

CAS를 활용하여 수학교육에 적용하는 방법의 하나로 단계별 문제해결의 모듈을 전문가시스템으로 확장하는 방법이 있다. 이것은 RME에 참여한 학생들이 계산기의 활용에서 경험한 어려움을 해소하는 방향으로 모듈이 개발될 수 있다(Drijvers, 1999). 예를 들면, 다양한 기호 연산에 관련된 문제해결 전략을 제공할 수 있고, 대수연산과 수치연산의 차이에서 나타나는 암시적 어려움을 미리 제거하는 모듈을 개발할 수도 있다. 그리고 교과서와 거의 일치하는 수식표현을 제공할 수도 있으며, 언제 어떻게 CAS를 활용할 것인지에 대한 학습자의 차이를 측정하여 개인별 학습을 촉진하는 시스템으로 확장이 가능할 것이다.

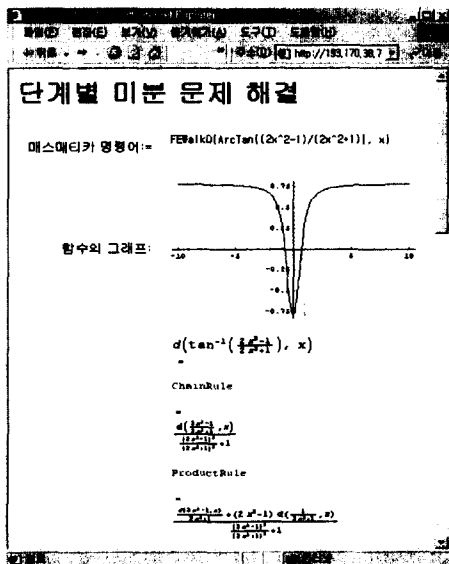
이런 부분과 관련하여 단계별 문제해결에 따른 단계 수를 측정하여 미적분 문제의 난이도로 추정하여 난이도에 따른 학습자의 차이를 고려한 학습시스템으로 발전할 수 있는 가능성이 있다고 판단된다(전영국, 1999).

한편, 학습자의 로그 기록을 DB화하여 그 학습자의 수학 지식을 관리해주는 시스템을 고려할 수 있다(The Omega group, 2000). 이것은 Mathematica의 커널을 가동시키는 로드를 줄일 뿐만 아니라 다양하고 광범위한 명령어의 사용에서 연유되는 인지적 부담을 줄이는 측면에서 중요하다. 즉, 학습자가 의도적으로 접근한 수식표현과 수학적 지식을 인덱싱하여 관리해줌으로써 관련있는 학습결과물을 하이퍼링크로

연결시키고, 개개인의 성향에 따라 지식관리의 인터페이스를 적응적으로 제공하게 된다. 이러한 학습관리 시스템은 사용자 모델링을 토대로 학습 결과물을 효율적이고 체계적으로 다시 접근함으로써 유연성있는 학습과 복잡도가 높은 수학적 사고를 용이하게 한다(Brusilovsky, 1996; Henze, Nejdil, & Wolpers, 1999).

3.3 J/Link를 이용한 단계별 미분 모듈의 구현

FEWalkD라고 명명된 이 모듈은 Mathematica 패키지로 다음과 같이 구현되었다. 이것은 현재 <http://www.calc101.com> 사이트를 개발 중인 George Beck이 개발한 패키지를 기초로 개발한 것이다. 패턴 매칭을 이용한 rewriting system을 구현하고, 각 단계의 결과를 웹상에서 표현하기 위하여 gridlist로 리스트 자료구조로 처리하는 것이 중요하다. 그 결과를 Mathematica 명령어인 GridBox로 처리한 다음 J/Link의 evaluateToOutputForm을 호출하고 그 결과를 Acme에서 제공하는 GifEncoder를 사용하여 stream 방식으로 웹서버에 전송한다. 부록에 있는 코드를 참고하면 FEWalkD에 대한 Mathematica 코드와 자바 서버릿으로 구현된 부분을 구조적으로 재구성할 수 있다.



(그림 3) ArcTan에 관련된 미분문제의 풀이 과정

4. 향후 방향

1) 문제 해결의 단계와 알고리즘 사용

미적분 문제의 해를 단계별로 제시하는 웹기반의 모듈은 학습자와 교사의 작업에 어떤 효과를 제공하는지에 대한 평가를 하는 것이 요구된다. 단계별로 제공하는 모듈의 개발은 일반적으로 어떤 특성이 있으며, 교수-학습의 어떤 단계에서 효과적으로 또는 효율적으로 활용될 수 있는지에 대한 연구도 필요하다. 일반적으로 컴퓨터대수 시스템을 학습 과정에 도입하는 것은 문제해결의 알고리즘을 지필계산으로 숙달한 후에 black box로 활용하는 것이 바람직하다. 기존의 알고리즘을 이해하고 계산하는 방법을 습득한 후에 새로운 데이터를 이용하여 예제를 다루어보고, 여러 가지의 예제를 통하여 문제의 속성을 파악하여 새로운 성질을 예측하고 증명하는 일련의 사이클에서 창조적인 수학학습이 이루어진다(Heugl, Klinger, & Lechner, 1996)

수학문제를 기호연산을 통하여 알고리즘화된 모듈은 black box의 형태로서 입력에 대한 출력만을 제공한다. 반면에 이 결과가 어떤 과정을 거쳐 도출되는지를 단계별로 제시하는 것이 white box이다(Buchberger, 1989). 학생들은 문제 해결의 투명한 과정을 통해 알고리즘을 이해하게 되고, 다시 procept를 통하여 새로운 문제를 해결할 수 있는 대상으로 구체화시킨다. 이런 일련의 과정이 앞서 소개한 창조적 사이클과 관련하여 CAS가 어떻게 활용되어야 하는가에 대한 연구는 거의 없다. 본 연구에서 개발한 모듈이 white box, 창조적 사이클, procept와 관련하여 어떻게 활용되어야 하는지에 대한 연구는 후속 과제로 남긴다.

2) 교사의 사용 방안

본 연구에서 설계하고 개발한 소프트웨어 모듈은 웹서버에서 구동되는 특성으로 인하여 사용자(학습자)와의 상호작용성이 단순하게 진행되는 단점이 있다. 이것은 문제해결의 각 단계별로 사용자가 다음 단계의 진행을 결정하거나 컴퓨터가 다음 단계를 진행하는 방식을 조절할 수 있는 모드(mixed initiative mode)를 지원하지 않는다. 이

러한 특성으로 인하여 교사는 교실에서 수업을 전개하면서 개념을 설명하는데 치중하고 각 단계에서 필요한 수식변환에 관련된 테크닉을 컴퓨터로 보여주는 도구로 활용할 수 있다. 그러나 이 소프트웨어 모듈이 학습용으로 더 발전하여 튜토리얼 또는 튜터 시스템으로 개발될 여지는 많다고 판단된다. 교사는 수업 중에 사용한 마우스 클릭의 패턴에 관련된 데이터는 웹서버의 로그 파일에 저장됨으로 서버 관리자의 입장에서 그 데이터를 나중에 분석할 수 있다. 이러한 데이터 분석에 관련된 모듈은 추가될 예정이다.

3) 웹 기반의 수식 표현

웹상에서 수식을 구현하는 부분은 MathML에 대한 표준화 작업과 MathML 브라우저의 개발이 수년 내로 이루어질 것이라는 예측에 따라 수학과 수학교육에 관련된 다양한 소프트웨어의 개발이 이루어질 전망이다. 참고로 J/Link를 사용하여 웹상에서 수식 표현을 구현하는 방법과는 별도로, PHP와 같이 웹서버에서 스크립트 형태로 처리하는 기법을 응용하면 server page 형태로 수식연산을 수행할 수 있다(박광덕, 1999). 현재 Mathematica 개발사인 Wolfram Research Institute에서 연구 중인 MSP(Mathematica Server Page)가 여기에 해당되는데 이것은 WebMathematica라는 소프트웨어로 개발 중이다. 이와 같은 여러 연구개발 동향을 감안하면, 컴퓨터대수 시스템의 기능을 활용하는 것이 중요한 관건으로 떠오르고 있는데, 특히 향후에는 컴퓨터대수 시스템을 수학지식을 관리하는 방법과 관련된 많은 연구가 요청되고 있다.

참 고 문 헌

- [1] 김영익, 이규봉(1999). 아름다운 수학 Mathematica와 함께. 교우사.
- [2] 박광덕(1999). PHP를 이용한 웹서버 프로그래밍. 삼각형.
- [3] 송기홍(1995). 수학적 개념의 형상화를 위한

- 컴퓨터 화면 구성. 대한수학교육학회논문집, 5(1), 55-64.
- [4] 전영국(1995). 일차방정식을 컴퓨터에게 가르쳐보는 과정을 통한 문제풀이: 시스템 개발과 형성평가. 교육공학연구, 11(2), 143-168.
- [5] 전영국(1997). 수학교육에서의 통신망 활용. 대한수학교육학회 논문집 7(1), 245-258.
- [6] 전영국(1997). WWW와 Mathematica를 이용한 CAI 개발. 대한수학교육학회 논문집 7(2), 281-292.
- [7] 전영국(1999). 지능형 교수시스템 개발을 위한 학습자 모델링 연구. 컴퓨터교육학회논문집, 2(2), 51-62.
- [8] 정상권, 추상목(1999). 수학교육에서의 Maple의 활용 방안. 대한수학교육학회지 학교수학 1(1), 157-186.
- [9] 황우형(1997). 그래픽 계산기의 중등수학교육 활용방안. 대한수학교육학회논문집, 7(2), 215-254.
- [10] Brusilovsky, P. (1996). Methods and techniques of adaptive hypermedia. *User Modeling and User-Adapted Interaction*, 6(2-3), 87-129.
- [11] Buchberger, B. (1989). Should students learn integration rules? Technical Report at RISC, Austria.
- [12] Buchberger, B. & Loos, R. (1982). Algebraic simplification. *Computer algebra: Symbolic and algebraic computation, Computing Supplementum 4*, 11-43. Springer-Verlag.
- [13] Drijvers, P. (1999). Students encountering obstacles using a CAS. Technical Report at Freudenthal Institute, The Netherlands.
- [14] Dubinsky, E. (1995). ISETL: A Programming Language for Learning Mathematics, *Comm. in Pure and Appl. Math.*, 48, 1-25.
- [15] Feurzieg, W. (1986). Algebra slaves and agents in a logo-based mathematics curriculum. *Instructional Science*, 14, 229-254.
- [16] Heck, A. (1993). Introduction to Maple. Springer-Verlag.
- [17] Henze, N., Nejd, W., & Wolpers, M. (1999). Modeling Constructivist Teaching Functionality and Structure in the KBS Hyperbook System. AIED '99 Workshop on Ontologies for Intelligent Educational Systems, Le Mans, France.
- [18] Heugl, H., Klinger, W., & Lechner, J. (1996). *Mathematikunterricht mit Computeralgebra-Systemen*. Bonn: Addison-Wesley.
- [19] Jun, Y. (1995). Learning how to solve linear equations by teaching the computer: Development and formative evaluation. Unpublished doctoral dissertation, University of Illinois at Urbana-Champaign.
- [20] Karian, Z. (Ed.) (1992). Symbolic computation in undergraduate mathematics education. MAA Notes 24, Mathematical Association of America.
- [21] Le, H. Q. (1999). Client-server communication standards for mathematical computation. Unpublished master thesis, University of Waterloo, Canada.
- [22] Laughbaum, E. (Ed.) (2000). Hand-held technology in mathematics and science education: A collection of papers. Ohio State University.
- [23] National Council of Teachers of Mathematics. (1989). Curriculum and evaluation standards for school mathematics. Reston, VA: NCTM.
- [24] Paul Libbrecht, P. (2000). ActiveMath. Talk given at Deduktionstreffen.
- [25] Tall, D. (Ed.) (1991). Advanced mathematical thinking. Kluwer Academic Pub.
- [26] The Omega group. (2000). Adaptive Course Generation and Presentation. ITS workshop on Adaptive and Intelligent Web-Based Education Systems.
- [27] Treffers, A. (1991). Didactical background of a mathematics program for primary

education. In L. Streefland (Ed.), Mathematics education in primary school: On the occasion of the opening the Freudenthal Institute. Utrecht: Center for Science and Mathematics Education.

- [28] Wester, M. J. (Ed.) (1999). Computer Algebra Systems. John Wiley.
- [29] Wolfram, S. (1991). Mathematica: A system for doing mathematics by computer. Addison-Wesley.

전 영 국



1986 수원대학교 수학과
이학사

1990 시카고주립대학교
수학과 이학 석사

1995 일리노이대학교 어바나
-삼페인 교육학 박사

1996~현재 순천대학교 컴퓨터교육과 조교수
관심분야: 지능형교수시스템, 웹기반학습시스템,
HCI

E-Mail: ycjun@rise.sunchon.ac.kr

부 록

FEWalkD 패키지에 해당되는 코드의 일부

```

FEWalkD[_, x_] := Module
{
  allrules, derivative, gridlist={},
  allrules = ToExpression[ListRules];
  derivative = D[f, x];
  (* Print[derivative, " ="]; *)
  (* gridlist=Append[gridlist, List[ToBoxes[derivative]]];
  gridlist=Append[gridlist, List[" ="]; *)
  gridlist=Append[gridlist, FEStyleExp[derivative]];
  gridlist=Append[gridlist, FEStyleStr[" ="];
  While
  {
    Not @ FreeQ[derivative, d],
    oldderivative = derivative;
    k = 0;
    While
    {
      oldderivative == derivative,
      k++;
      derivative = derivative /. allrules[[k]]
    };
    (* Print[k]; *)
    (* Print[ListRules[[k]] ]; *)
    gridlist=Append[gridlist, FEStyleStr[" "]];
    gridlist=Append[gridlist, FEStyleStr@ListRules[[k]]];
    gridlist=Append[gridlist, FEStyleStr[" "]];
    (* Print["= ", derivative]; *)
    gridlist=Append[gridlist, FEStyleStr[" = "]];
    gridlist=Append[gridlist, FEStyleExp[derivative]];
  };
  (* D[f, x] *)
  gridlist=Append[gridlist, FEStyleStr[" "]];
  gridlist=Append[gridlist, FEStyleStr[" Simplify "]];
  gridlist=Append[gridlist, FEStyleStr[" = "]];
  gridlist=Append[gridlist, FEStyleExp[Simplify[D[f, x]]];
  gridlist = Map[List, Flatten[gridlist]];
  GridBox[gridlist, ColumnAlignments->Left]
}

```

/* J/Link로 구현한 자바서브릿 코드의 일부 */

```

import com.wolfram.jlink.*;
import com.wolfram.jlink.MathLinkFactory;
import com.wolfram.jlink.MathLinkException;
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.awt.*;
import java.awt.image.*;
import java.net.URLEncoder;
import Acme.JPM.Encoders.GifEncoder;

public class WalkD extends HttpServlet
{
  static final String kernelCommandLine =
    "-linkmode launch -linkname 'C:\\ntca4\\MathKernel.exe
-mathlink'";
  Frame frame = null;
  Graphics g = null;
  Image outputImg;
  byte[] outputBytes;
  String errStr;
  int maxX = 400;
  int maxY = 300;

  public String doComputation(String inputStr, PrintWriter out) {
    KernelLink ml;
    String outputStr = new String("");

```

```

try {
  ml = MathLinkFactory.createKernelLink(kernelCommandLine);
  ml.connect();
  ml.discardAnswer();
  ml.evaluate("<<<FEWalkD.m");
  ml.discardAnswer();

  outputStr = new String(ml.evaluateToOutputForm(inputStr, 0));
  ml.close();
} catch (MathLinkException e) {
  out.println("MathLinkException: " + e.toString());
}
return outputStr;

```