

Mathematica 소개

민 경 원*

1. 머릿말

우리는 간단한 수학 계산을 하기 위하여 전자 계산기를 이용하며 복잡한 계산이나 반복적인 계산을 수행하기 위해서는 일련의 계산과정을 종합한 프로그램을 만들어 컴퓨터를 사용하기도 한다. 그리고 계산 결과를 확인하기 위하여 그래픽으로 변환하기도 한다.

그러나 이러한 과정은 프로그램을 만들어 수행하여 본 사람은 반드시 느꼈을 사항으로 매우 불필요한 시간을 요하는 낭비적인 작업이라 할 수 있다. 예를 들면 계산 과정을 설명하여 주는 flow chart를 구성한 이후에도 적절한 서브루틴을 선택하여 주프로그램에 접목시켜야 하며 input 과 output 자료의 정리, 프로그램의 여러 수정 작업, 그리고 수치결과의 그래픽 작업도 귀찮은 일이라 할 수 있겠다.

또한 기존의 프로그래밍 언어는 수치 결과만을 제공하여 주기 때문에 수학적 모델의 전개 과정에 많은 제한이 있어 왔으며 변수의 변화에 따른 결과의 다양성을 볼 수 없어 연구의 진행에 한 장애물이 되어 왔었다.

따라서 위의 한계를 극복할 수 있는 새로운 개념의 프로그래밍 언어에 대한 필요성이 대두되었고 최근 널리 이용이 되고 있는 소프트웨어가 바로 'Mathematica'인 것이다.

Mathematica는 이미 미국에서는 각 대학이나

연구 기관에서 여러 분야의 연구에 널리 이용이 되고 있으며 기능의 풍부함과 사용의 편리함으로 인하여 사용자가 급속히 확산되고 있다. 국내에서도 몇년 전부터 개인적으로 쓰기 시작하여 현재 수학에 관련된 분야에서 많이 이용을 하고 있다.

그러나 국내에서의 Mathematica에 대한 대부분 사용자의 인식은 기능이 풍부한 계산기에 머물고 있어 단순한 계산을 수행하기 위하여 이용하고 있을 뿐이다. 따라서 본 난을 빌려 Mathematica의 올바른 이해와 사용을 위하여 간단한 소개와 더불어 기능 중심으로 예제를 들어 설명하기로 한다.

2. Mathematica란 무엇인가?

Mathematica를 간략하게 말하면 "수학을 컴퓨터로 할 수 있게 해주는 고성능 수학패키지이며 프로그래밍 언어"라고 설명할 수가 있다. 1988년 6월 애플 매킨토쉬 컴퓨터용으로 버전 1.0이 공식적으로 발표되었으며 계속 보완이 되어 현재 버전 2.0(1991년 1월 발표)이 다음과 같은 컴퓨터 시스템용으로 이용이 되고 있다. CONVEX, DEC VAX (Ultrix와 VMS), RISC, Hewlett-Packard/Apollo, IBM 386 PC(MS-DOS와 Microsoft Windows), IBM RISC, MIPS, NeXT, Silicon Graphics, Sony, Sun(SPARC compatible).

Mathematica는 1988년 발표 이후 수학을 이용한 연구의 새로운 장이 열렸다고 여러 시사지와

* 정희원, 삼성종합건설 기술연구소, 선임연구원

전문지에서 평가하였으며 그해에 여러 상을 수상한 바도 있다. 특히 Business Week 잡지 선정 1988년 세계 10대의 최고 개발품으로 뽑히기도 하였다. 이후 Mathematica에 관한 저널지가 매월 발간되고 있으며 매년 사용자들이 모여 학술발표회를 개최하기도 한다.

Mathematica는 자연과학이나 공학에서 발생하는 제반 문제가 일단 수학적 모델이 되면 다음과 같은 기능을 이용하여 해결을 해주는 소프트웨어이다.

첫째, IMSL 또는 MATLAB 등에서와 같이 수치적으로 계산하여 결과를 구할 수도 있고

둘째, MACSYMA 또는 MAPLE 등에서와 같이 기호로서 계산하여 결과를 기호로 얻을 수 있으며

셋째, 결과를 MATLAB 또는 MATHCAD에서 처럼 그래픽으로 처리하여 이해를 용이하게 할 수가 있다.

넷째, 기존의 언어인 BASIC, C, FORTRAN 처럼 일련의 계산과정을 Mathematica언어를 이용하여 프로그램할 수가 있다.

마지막으로, 인터페이스 기능이 마련되어 있어 다른 프로그램과의 자료 교환이 가능하며 Mathematica에 내장된 함수를 외부에서 서브루틴으로 이용할 수가 있다.

3. Mathematica의 기능과 예

Mathematica를 이용하는 가장 간단하고 초보적인 방법은 계산기로써 사용하는 것이다. 계산과정을 입력하면 Mathematica는 결과를 즉시 보여주고 그래픽으로 변환해 주기도 한다. 따라서 초보자의 입장에서는 Mathematica를 훌륭한 계산기라는 도구로 우선 친숙하여 질 필요가 있다. 이후 이용자의 요구가 늘어나게 되며 이때에 Mathematica의 필요한 기능을 이용하여 최종적으로 프로그래밍 단계까지 이르면 되는 것이다.

Mathematica는 일반적인 컴퓨터 시스템이 지원하는 내장된 30개의 수학 기능을 가지고 있는데 비하여 약 750개 이상의 것을 가지고 있다. 따라서 서브루틴을 별도로 이용할 필요가 없게 되는 장점이 있다. 한 예로 매트릭스의 연산 기능이

내장되어 있어 명령어으로써 간단히 이용할 수 있다. 또한 수치 계산외에 기호 처리와 그래픽 처리가 가능하여 기호가 있는 식의 전개는 물론 근도 쉽게 구할 수가 있다.

위와 같은 기능을 다음과 같은 간단한 예를 들어 설명하기로 하자. Mathematica는 간단한 '대화식'으로 되어 있어 In[n]:=이라는 곳에 수행하고자 하는 자료 또는 명령어를 보내면 즉시 그 결과가 Out[n]:=이라는 곳에 나타난다.

3.1 수치 계산 기능

예: $\log(4\pi)$ 의 수치값을 구하라.

```
log[4 Pi]는 Mathematica로 된 In[1]:=N(Log[4 Pi])
입력 형태이며 N은 수치값을 구하
라는 명령어이다.
결과가 즉시 나타난다. Out[1]:=2.53152
수치적인 정밀도를 향상시킬 수 In[2]:=N(Log[4 Pi], 40)
있으며 한 예로 다음과 같이 40 Out[2]:=2.53102424696929079
자리까지 표현할 수 있다. 2977891594269411847798
```

기존의 계산기나 컴퓨터 시스템은 단지 지정된 정도의 정밀도만을 고려하는데 Mathematica는 어떠한 정밀도도 다룰 수가 있다. 또한 다음과 같은 복잡한 수학기능도 다룰 수가 있다. elliptic integrals, complex Bessel functions, hypergeometric functions, integer factorization 등등.

또한 Mathematica는 매트릭스의 처리에 관한 기능이 내장함수로 되어 있어 역 매트릭스, 고유치, 고유벡터 등에 관한 계산도 명령어으로써 간단히 처리할 수 있다. 그리고 수치적분, 수치최소화, 선형 프로그래밍의 처리가 가능하며 대수방정식, 평미분방정식의 해도 구할 수가 있다.

3.2 기호 계산 기능

예: 다음과 같은 적분을 행하여라.

$$\int x^4(x^2-1) dx$$

```
다음은  $x^4(x^2-1)$ 의 Mathematica In[1]:= $x^4(x^2-1)$ 
형태이다.
Out[1]:= $-\frac{x^4}{1+x^2}$ 
다음은 바로 전의 표현을 적분하라 In[2]:=integrate[%, x]
는 명령어이다.
Mathematica는 다음과 같은 정적
분 결과를 보여준다.
Out[2]:= $x + \frac{x^3}{3} + \frac{\text{Log}[-1+x]}{2}$ 
 $-\frac{\text{Log}[1+x]}{2}$ 
```

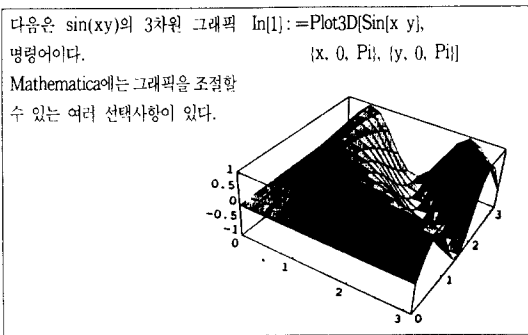
Mathematica가 다루는 기호처리 능력 중에서 핵심중의 하나는 대수식을 다룰 수가 있다는 점이다. 예를 들면 복잡한 다항식을 인수분해를 할 수 있으며 역으로 전개를 할 수 있을 뿐만 아니라 이와같은 과정을 통하여 간단히 만들 수가 있다. 또한 다항식으로 이루어진 방정식의 해를 구할 수가 있다.

기호처리 기능을 이용하면 기존의 컴퓨터 시스템으로는 유도하기 힘든 수학적 식의 유도가 가능하다. 필자의 경험으로는 기호로 표시된 매트릭스 연산 수행과 다항식의 근을 유도하여 구조시스템의 고유치해석에 대해 변수연구를 수행한 바 있다.

더구나 기호로 된 식의 미분과 적분이 가능하며 상미분방정식의 해도 구할 수가 있다. 또한 멱급수의 유도, 처리, 극한도 다룰 수가 있으며 벡터 해석과 라플라스변환도 처리할 수 있다.

3.3 그래픽 기능

예 : 함수 $\sin(xy)$ 를 $0 < y < \pi$ 의 범위내에서 그려라.



Mathematica는 2차원, 3차원 그래픽을 다룰 수 있으며 등고그림 및 밀도그림도 나타낼 수 있다. 그리고 데이터의 그래픽뿐만 아니라 함수의 그래픽도 쉽게 다룰 수가 있다. 또한 그래픽 결과를 조정할 수 있는 여러 기능을 가지고 있는데 예를 들면 음영, 색깔, 조도 등이 있으며 애니메이션 기능도 있다.

3.4 Programming 언어 기능

Mathematica를 이용하여 수학적으로 어려운 문제를 해결하고자 할 때, 하나의 명령어만을

입력하여 결과를 얻는 단순한 작업보다는 여러 명령어를 결합하는 경우와 동일한 계산 과정을 상이한 입력으로 인하여 여러번 반복적으로 하여야만 하는 경우가 있다. 바로 이러한 때에 프로그램이 필요한 것이며 Mathematica가 이것을 쉽게 해결해 줄 수 있는 것이다.

Mathematica는 프로그래밍 언어로서 BASIC, C, FORTRAN과 같은 기존의 언어와는 달리 수치 처리 기능외에도 기호 처리 기능 및 그래픽 기능을 수행할 수 있는 새로운 형태의 언어이다. 또한 Mathematica는 약 750개 이상의 내장함수 기능과 별도의 수학패키지가 마련되어 있기 때문에 복잡한 문제를 수월하게 해결할 수 있다. 즉 내장함수가 많아 별도로 서브루틴을 접속시킬 필요가 없어 프로그래밍 하기가 수월하며 원하는 변수의 출력을 변수만 쓰면 할 수가 있기 때문에 에러 수정이 매우 수월하다. 또한 결과의 그래픽 기능이 있기 때문에 손쉽게 결과를 확인할 수 있다.

따라서 Mathematica를 이용하여 프로그램을 만들려면 프로그래밍 언어로서의 규칙들을 익혀야 한다. 그러나 Mathematica의 언어는 기존의 BASIC, C, FORTRAN과 거의 비슷하기 때문에 간단한 규칙만을 습득하면 사용하기에 큰 어려움은 없다. 여러분은 종이와 연필을 가지고 문제를 푸는 입장에서 단지 Mathematica 하나만 가지고 문제를 해결하는 입장에서 있는 것이다.

예 : 1에서 5까지의 자연수의 제곱을 프린트하라.

```
Do 루프를 이용하였으며 결과가 In[1]:=Do[Print[i^2], {i, 1, 5}]
즉시 프린트되었다.      1
                          4
                          9
                          16
                          25
```

예 : 삼각함수의 곱 형태로 표시된 식을 더하기 형태로 변환시켜 주는 법칙을 설정하고 예를 들어 설명하라.

다음은 변환 법칙을 설정하는 프로그램이다.

```
In[1]:=TrigLinearRules={
  Sin[x_] Sin[y_] :> Cos[x-y]/2-Cos[x+y]/2,
  Cos[x_] Cos[y_] :> Cos[x-y]/2+Cos[x+y]/2,
  Sin[x_] Cos[y_] :> Sin[x-y]/2+Sin[x+y]/2,
}
```

다음은 위의 법칙을 이용하여 곱의 형태를 더하기 형태로 변환하는 예이다.

```
In[2]:=Sin[a] Cos[b]+Sin[a] Cos[a]+Cos[2a] Cos[3a]/
  TrigLinearRules
Out[2]:=Cos[a] Cos[a] + Cos[5a] Cos[a] + Sin[2a] Cos[a] + Sin[a-b]
```

예 : 소수를 표현하기 위한 함수를 만들어라.

```
다음은 n개의 소수를 만들기 위한 In[1]:=f[n_]:=Table[Prime[i]
  함수 f를 정의하는 것이며 내장된 , {i, n}]
  식인 Prime[ ]을 이용하였다.
  정의된 함수 f를 다음과 같이 즉시 In[2]:=f[10]
  이용할 수 있으며 처음부터 10개까 Out[2]:= {2, 3, 5, 7, 11,
  지의 소수가 결과로써 나타난다. 13, 17, 19, 23, 29}
```

3.5 인터페이스 기능

Mathematica 시스템은 연산을 처리하는 kernel 과 사용자와 상호작용하는 front end의 두개 부분으로 되어있다. kernel은 컴퓨터에 관계없이 동일하나 front end는 컴퓨터 기종에 따라 GUI(graphical user interfaces)가 실현되도록 되어있는 notebook이라는 진보된 기능을 지원하고 있다. 예를 들면 윈도우 버전의 Mathematica를 이용하면 notebook기능을 이용하여 편집을 용이하게 할 수 있다.

또한 Mathematica는 다른 프로그램과의 자료 교환을 위하여 일반적인 소프트웨어기준을 따르고 있는데 예를 들면 그래픽은 PostScript로 되어있어 그래픽을 출판이나 다른 프로그램에 접목할 수가 있다. 또한 Mathematica는 C, Fortran 또는 TeX 같은 시스템으로 된 자료를 읽을 수 있을 뿐만 아니라 출력을 할 수가 있다.

Mathematica는 MathLink를 이용하여 다른 프로그램과의 통신이 가능하다. 다른 프로그램은 MathLink에 적합하게 변환이 되어 Mathematica와 자료 교환 및 명령어 공유가 가능하다. 이러한 MathLink를 이용하여 외부 프로그램에 Mathematica를 접목시킬 수가 있다. Mathematica 언어를 이용하여 입력과 명령어를 만들고 이것을

MathLink를 이용하여 외부 프로그램에 보내는 것이다.

4. 맺는 말

이상과 같이 Mathematica의 고유한 특징과 많은 기능 중의 일부분만을 예를 들어 설명을 하였다. 그러나 Mathematica의 피할 수 없는 단점은 많은 수학적 기능이 포함되어 있기 때문에 처리 속도가 늦다는 점이다. 예를 들면 많은 량의 반복 작업이나 차수가 큰 매트릭스의 연산 작업은 다소 속도가 늦어 PC기종에서는 곤란을 겪을 때가 많다. 따라서 PC 대신 workstation같은 상위 기종의 컴퓨터를 이용한다면 처리 속도가 빨라져 진행에 문제점이 없다. 한 예로 workstation에서는 차수가 30개인 고유치 해석도 내장함수인 명령어만으로 단지 몇초만에 할 수 있는 데 비하여 PC에서는 기종에 따라 몇배, 몇백배의 시간이 요구되는 것이다. 그리고 또 하나의 단점으로는 방대한 프로그램을 운용하기 위한 비용(ram)이 많이 든다는 점이다. 한 예로 PC에서는 기본적으로 Mathematica를 작동하기 위해 최소한 4 mega ram이 필요하며 여러 수학적 기능을 충분히 이용하기 위해 많은 량의 ram이 필요하다는 점이다.

그러나 위의 단점은 Mathematica가 지니고 있는 고유한 장점을 생각한다면 매우 미미한 것이라 여겨진다. 수학의 대부분의 기능을 포함하고 있으며 기호처리가 가능하고 프로그래밍 기법이 다양하기 때문에 수학을 이용하여 연구를 하는 사람에게는 훌륭한 도구가 생긴 것이라 할 수 있다.

참 고 문 헌

- [1] Wolfram, S., *Mathematica: A System for Doing Mathematics by Computer*, Second Edition, Addison-Wesley, 1991.
- [2] Maeder, R., *Programming in Mathematica*, Second Edition, Addison-Wesley, 1991.
- [3] Noor, A.K., Elishakoff, I, and Hulbert, G. *Symbolic Computations and Their Impact on Mechanics*, PVP-Vol.205 ASME, 1991.