

# CAS(Computer Algebra System)와 Symbolic Calculation

강 상 균 <서울대학교 조선해양공학과 4학년>, 신 종 계 <서울대학교 조교수>

## 1. CAS (Computer Algebra System) 란?

이제 공학의 제반분야에서 컴퓨터의 이용이 필수적이라는 점에 대해 이의를 제기할 사람은 아무도 없을 것이다. 초창기의 계산기 역할에서 벗어나, 이제는 설계를 위한 도구로서 자리잡고 있다. 하지만, 요사이 이루어지는 컴퓨터의 비약적인 성능 발전에 비해 그 활용의 범위나 방법이 예전에 비해 크게 진전이 있다고 보기는 어렵다. 비록 하루가 다르게 새로운 기종 및 혁신적인 소프트웨어가 등장하고 있지만 실제 산업계나 연구기관, 학계에서 그러한 발전들을 쉽게 받아들여 이용하지 못하고 있다. 그 현상은 특히 소프트웨어부분에서 두드러져 보이는데, 그것은 아마도 많은 사람들이 새로운 소프트웨어에 대해 알 수 있는 기회가 적거나 다행히 어떤 굉장한 소프트웨어에 대해 들었다 하더라도 시간과 돈을 들여 그것을 새로 배우는 것이 정말로 이득이 되는지 확신이 서지 않기 때문일 것이다. 따라서, 현재는 앞선 것이지만 곧 구식이 되어버릴지도 모르는 소프트웨어의 사용법을 익히는데 투자하느니 예전부터 사용해와서 익숙해진 도구를 조금 불편하더라도 참고 쓰게된다. 예를 들면 식의 유도 및 전개 등은 손과 연필을 사용하고 간단한 계산은 계산기나 LOTUS같은 spreadsheet를 사용하고, FORTRAN으로 수치해석 프로그램 짜고, 그 결과는 또 다른 그래픽 처리 프로그램을 사용하는 등이다.

공학문제를 해결해 나가면서 위에 말한 과정들이 유기적으로 통합되어야 작업효율이 높아질 것임은 말할 나위가 없다. 하지만 현실은 그런 이상에서 많

이 동떨어져 있는 것이 사실이다. 그러나, 이제는 그러한 상황을 바꿀 수 있는 도구(소프트웨어)들이 많이 등장하여 공학분야에 종사하는 사람들의 작업 방식을 혁명적으로 변화시킬 수도 있는 단계에 이르렀다.

구체적으로 이야기하자면, 단순히 '계산'을 해주는 것 뿐만이 아니라 이전에는 사람이 손으로 직접 해야만 했던, 문자가 들어가는 식에 대한 연산들(예를 들어 부정적분, 미분, 식의 전개, 인수분해, Laplace 변환, Fourier 변환 등) 및 수치해석, 그리고 그 결과에 대한 그래픽 처리 기능을 한군데 통합한 소프트웨어들이 등장하고 있는 것이다. 특히, 숫자가 아닌 문자를 하나의 독립된 개체로 다루어 이에 대하여 여러 가지 수학적 연산을 하는 것을 symbolic calculation이라 한다. 이와 같이 symbolic calculation과 수치계산(numeric calculation), 그래픽처리(graphic processing) 등을 한꺼번에 할 수 있는 소프트웨어를 Computer Algebra System (CAS)라 한다.

CAS중 현재 주로 쓰이는 것으로는 Mathematica, Macsyma, Maple, AXIOM, Derive 등을 들 수 있다. 이들은 대개 고유의 user interface 및 언어를 갖고 있으며, symbolic calculation외에, 막강한 수치연산(numeric calculation) 및 그래픽(graphic) 기능들도 보유하고 있다. 또한, 이들의 기능상 차이는 그다지 많지 않으므로 그중에 어느 것을 선택하는가 하는 것은 주로 외관, 가격, 주위 사람들의 평가 등에 좌우된다. 그중에 많이 쓰이는 것은 Mathematica와 Maple V이다.

이런 소프트웨어를 사용하면 복잡하고 자질구레한 작업에 드는 시간과 노력을 아낄 수 있어 주어진 문제를 파악하고 그 해결책을 찾아내는데 더 집중할 수 있게 된다. 즉 더욱 더 창조적인 일에 몰두할 수 있게 해준다. 하지만 이런 소프트웨어들은 그 기능이 막강한 대신에 능숙하게 사용하기 위해서는 많은 시간과 노력이 요구되는 단점이 있다.  $\int x^6 \cos(3x) \sin(2x) dx$  같이 복잡한 문제라면 몰라도 간단한 계산문제를 풀기위해 몇시간동안 매뉴얼을 붙잡고 있어야 곤란할 것이다.

정통적인 CAS는 기능이 강력한 대신 사용이 어렵다는 것을 앞서 말했다. 좀 더 쉬운 것으로는 MathCAD라는 것이 있다. 이것을 사용하여 계산한 것을 보면 우리가 손으로 계산한 것과 매우 비슷한 상태로 입력과 출력이 나타나는 것을 알 수 있다. 즉, 직관적인 user interface를 가지고 있다는 뜻이다. 하지만 MathCAD는 그 기능이 여타 CAS에 비해 특히, symbolic calculation 분야에서 미약하고 주로 수치 계산에 이용된다.

그리고, 공학도로서 알고 있어야 할 소프트웨어로서 MatLab이라는 것이 있다. 이 소프트웨어는 행렬 연산에 기반을 두고 있다. 주로 사용되는 분야는 신호처리 (signal processing), 디지털 통신 (digital communication), 제어 시스템 설계 (control system design) 등이며 역시 symbolic calculation보다는 numeric calculation을 기본으로 한다. 하지만 toolbox라는 것을 이용하여 Maple V의 symbolic calculation기능을 연결하여 사용할 수 있다. 또한 Mathematica와도 연결하여 사용이 가능하다. MatLab 역시 기능이 다양하고 방대한 소프트웨어

이어서 배우는데 시간과 노력이 많이 든다.

주로 사용되는 CAS (Mathematica, Maple V) 와 MathCAD, MatLab에 대한 비교를 <표1>에 실는다[8] (1995. 가을 현재).

다음 절에서는 CAS의 주요 기능 중 가장 돋보이는 symbolic calculation에 대해 더 자세히 알아본다.

## 2. Symbolic Calculation이란?

다음과 같은 2차 방정식을 풀라는 문제가 주어졌을 경우,

$$3x^2 - 4x + 1 = 0 \quad (1)$$

그 답은

$$x = \frac{1}{3}, 1 \quad (2)$$

이 고 수치해석 방법을 따르면 그 답은

$$x = 0.3333, 1.0 \quad (3)$$

이 된다. 이것이 어떤 차이가 있느냐 하면,  $x = \frac{1}{3}, 1$  은 정확한 값인데 반하여  $x=0.33333, 1.0$  중에,  $x = 0.33333$ 은 근사값이라는 점이다. 또한 일반적인 이차방정식과 그 해는 아래와 같다.

$$ax^2 + bx + c = 0 \quad (4)$$

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (5)$$

계다가 (4)와 같은 문제는 전통적인 수치해석적

<표 1>

	Mathematica	Maple V	MathCAD	MatLab
현재 버전	2.2	3.0	6.0과 6.0 Plus	4.2
운영 체제	DOS, Windows, OS/2, Mac, Unix	DOS, Windows, Mac, Unix	DOS, Windows, Mac, Unix	Windows, OS/2, Mac, Unix
최소 하드웨어 사양 (Windows version)	386, 4M RAM, 13M disk, 16M swap	386, 4M RAM, 12 M disk	386, 8M RAM, 20M disk, 12M swap	486, 8M RAM, 15M disk
가 격	\$995	\$795	\$130 (6.0) \$350 (6.0 plus)	\$1695
비 고	연구기관에서 널리 사용됨	교육기관에 많이 사용	Windows가 주된 운영체제	Add-on toolbox가 다양하다.
관련 자료	Wolfram Research Inc. (800) 441-6284(미국) info@wri.com mathsource.wri.com http://www.wri.com	Waterloo Maple Software Inc. (800) 267-6583(캐나다) info@maplesoft.on.ca ftp.maplesoft.on.ca http://www.maplesoft.com/Maple	MathSoft Inc. (800) 628-4223(미국) mathsoft.mathsoft.com http://www.mathsoft.com	The MathWorks Inc. (508) 653-1415(미국) info@mathworks.com ftp.mathworks.com http://www.mathworks.com

방법으로는 풀 수가 없다. 아직도, 문자가 들어가는 식의 계산은 상당부분이 종이와 연필의 몫이 되고 있는 실정이다.

Symbolic calculation이라는 것을 쉽게 이야기하자면 컴퓨터를 써서 (4)식과 같은 문제를 풀어서 (5)식과 같은 답을 얻고, (1)식과 같은 문제도 답을  $x = 0.33333$ , 1.0 대신에  $x = \frac{1}{3}$ , 1으로 얻어내자는 것이다. 앞서 제시한 문제는 간단하니까 손으로도 푸는데 별 문제가 없지만 변수도 여러개고 차수도 높다면 사정이 달라진다. 항이 수십개씩 되는 식을 전개하고, 적분하고 미분하는 일을 생각해 볼 때, 게다가 계수가 복잡하기라도 하면 웬만해서는 아예 포기하게 될지도 모를 일이다.

### 3. Symbolic Calculation의 활용

아래와 같은 식을 적분해야 한다고 생각해 보자.

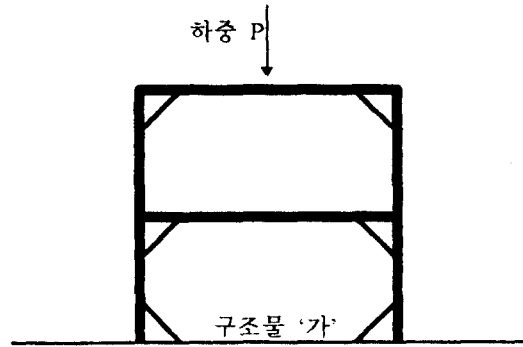
$$f(x) = x^6 \cos(3x) \sin(2x)$$

이 식이 적분된다는 것은 누구나 다 알 수 있다. 하지만 막상 손으로 하려니 너무 복잡할 것 같다. 이런 경우 symbolic calculation을 해주는 소프트웨어를 이용하여 다음과 같이 적분할 수 있다.

$$\begin{aligned} \int x^6 \cos(3x) \sin(2x) dx &= -360(x) + 180x^2 \cos(x) - 15x^4 \cos(x) \\ &+ \frac{x^6 \cos(x)}{2} + \frac{72 \cos(5x)}{15625} - \frac{36x^2 \cos(5x)}{625} \\ &+ \frac{3x^4 \cos(5x)}{25} - \frac{x^6 \cos(5x)}{10} - 360x \sin(x) \\ &+ 60x^3 \sin(x) - 3x^5 \sin(x) + \frac{72x \sin(5x)}{3125} \\ &- \frac{12x^3 \sin(5x)}{125} + \frac{3x^5 \sin(5x)}{25} + c \end{aligned}$$

그러면 이런 기능들은 공학문제에 어떻게 이용될 수 있을 것인가?

지금까지 컴퓨터를 사용하여 공학문제를 푸는 가장 일반적인 방법은 수치해석기법을 사용하는 것이다. 이런 기법들을 사용하여 사람이 직접 손으로 할 수 없던 많은 일을 할 수 있게 되었고 제반 분야에서 실로 엄청난 발전을 이룩하게 되었다. 하지만 수치해석적인 기법들도 단점이 있다. 가장 큰 단점은 수치해석법은 구체적인 예에만 적용된다는 것이다. 즉, 주어진 문제가 문자가 아니라 숫자로 표현되어야 한다. 처음부터 계산 자체가 안되는 것보다야 훨씬 낫



지만 그 각각의 결과에서 어떤 경향성을 발견하기가 쉽지 않기 때문에 설계에 적극적으로 활용하는데 한계가 있다.

예를 들어, 위와 같은 구조물을 설계하고자 할 때, 주어진 하중에 의해 각 구조부재에 걸리는 응력을 계산한다. 구조물이 복잡하면, 수치해석법을 이용하게 되는데 기존의 수치해석법은 숫자를 입력하여야만 계산이 가능하다. 즉, 길이, 단면, 폭과 높이, 탄성계수 등이 구체적으로 주어져야만 응력이 계산될 것이다. 그렇게 계산된 응력이 예를 들어,  $10\text{kN/cm}^2$  이라 하자. 이 값은 입력한 내용들이 어떤 효과를 주는 지 알 수 없는 수치일 뿐이다. 계산 결과가 부재치수와 탄성계수의 문자식으로 표현될 수 있다면 설계자의 입장에서는 훨씬 빠르고 정확하게 설계치수를 결정할 수 있을 것이다. 이와 같이 실제 공학분야에서 문자(symbol)로 미분, 적분, 기타 연산을 하여 문자로 설계식을 얻을 수 있다면 그 결과는 훨씬 유용할 것이다.

### 4. Symbolic Calculation의 주요 기능 및 응용 현황

Symbolic Calculation은 주로 다음과 같은 기능을 포함한다.

Symbolic Operation : 식의 정의, 함수의 정의, 기타 symbolic 연산  
 식의 정리 : 식의 전개, 인수분해, 약분, 통분 등  
 행렬 다루기 : 전치행렬, 역행렬, 행렬식, 고유치 등  
 방정식 풀기 : 다항방정식, 연립방정식, 미분방정식 등

CAS가 행하는 주요 기능 (symbolic calculation, numeric calculation, graphic processing) 중, symbolic calculation은 매우 흥미롭다. CAS가

이렇게 일견 신기해 보일 수 있는 이런 작업들을 해 낼 수 있는 것은 많은 경우 expert system과 pattern matching기술에 기반된 것이며 LISP계열의 언어들이 이런 목적에 가장 잘 맞기 때문에 그들의 언어는 대개 LISP에 기반을 두고 있다.

이 symbolic calculation은 현재 많은 공학적인 문제에 그 적용이 시도되고 있다. 복잡한 tensor 계산을 처리하는 것이 그 처음 적용중의 하나였으며 (Howard, 1980) 그 외에도 random vibration 과 stability, machine dynamics, accoustics and fluid mechanics, perturbation and bifurcation theory, robot dynamics, wave propagations in poly crystals 그리고 variational method 등에 적용되어 왔다[1].

특히, FEM (Finite Element Method)는 그 활용이 기대되는 분야인데 이미 Cecchi와 Lami에 의해 강성행렬의 자동생성에 대한 연구가 이루어졌고 그 이후에도 많은 연구가 이어졌으며 Beltzer는 고전적인 변분법(variational method)과 FEM을 symbolic calculation을 통하여 수행 하는 것에 대해 다루고 있다[2]. 근래의 연구 내용으로는 Leff와 Yun이 유한요소가 shape parameter의 함수로 이루어지는 전체강성행렬을 생성하는 시스템을 개발하여 16node, 33 element로 이루어진 truss에 대한 symbolic해석을 수행하였다[3]. Ioakimidis는 symbolic calculation에 series solution 및 iterative method를 적용하여 속도를 높이는 시도[5] 및 Gröbner basis[7]를 이용하여 간단한 truss 구조물의 force equation, reaction equation 및 compatibility equation의 자동 유도[6] 등을 연구하였다.

## 5. Mathematica 사용 예

이제, 앞서 열거한 CAS 중 Mathematica를 사용한 예를 들어본다. Mathematica는 Stephen Wolfram의 Wolfram Research사에서 만든 symbolic calculation 패키지로서[9][10] 1988년 처음 발표되었으며, 1995년 8월 현재 Apple Macintosh, MS-Windows, DOS 및 다양한 Unix용 버전이 판매되고 있다.

아래의 보기들은 Windows용 Mathematica 2.2를 이용하여 입력과 출력을 얻어낸 것이다.

### (1) 미적분

$f(x) = e^x \sin(x) \cos(x)$   
를 정의하여 그것을 적분해 보자.

```
expr = E^x Sin(x) Cos(x);
Integrate(expr, x) (*expr을 x에 대하여 부정적분한다.*)
```

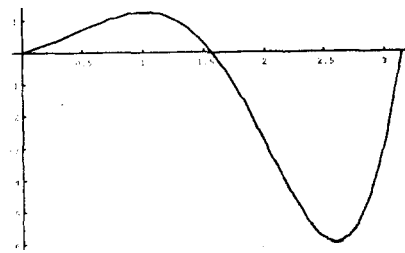
$$\frac{-2 E^x \cos(2 x) + E^x \sin(2 x)}{10}$$

```
Integrate(expr, {x, 0, Pi}) (* 정적분 *)
```

$$\frac{1}{5} - \frac{E}{5}$$

부정적분의 결과가 x를 포함한 식으로 나왔고, (0,  $\pi$ )사이의 정적분 결과도 근사값 -4.428이 아닌 정확한 값  $\frac{1}{5} - \frac{e^x}{5}$ 으로 출력되었다. 방금 계산한 결과를 그래프를 통하여 다음과 같이 근사적으로 확인할 수 있다.

```
Plot(expr, {x, 0, Pi})
```



-Graphics-

그래프에서 x축 아래의 면적이 x축 위의 면적보다 크므로 적분값은 음임을 알 수 있다. 이와 같이 계산과 그래픽 처리가 한데 통합되어 생산성을 높여주는 것이 CAS의 장점이다.

### (2) 행렬연산

대개의 symbolic calculation 패키지들이 행렬연산 기능을 갖추고 있다. Mathematica는 행렬  $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ 를 {{a, b}, {c, d}}로 표시한다. 행렬의 전치(transpose), 역행렬 구하기, 행렬식, 행렬의 고유치, 고유벡터 구하기 등을 주요 기능으로 갖고 있다.

행렬  $\text{mat} = \begin{pmatrix} 1-s & 2 \\ 4 & 5-s \end{pmatrix}$  의 행렬식 및 고유치, 고유값은 다음과 같이 구할 수 있다.

```

mat = {{1 - s, 2}, {4, 5 - s}}; (*행렬을 정의*)
Det(mat) (*행렬식*)
2
-3 - 6 s + s
Eigensystem(mat) (*고유치 및 고유벡터*)
{{3 - 2 Sqrt[3] - s, 3 + 2 Sqrt[3] - s},
 {{-1 - Sqrt[3]}/2, 1}, {-1 + Sqrt[3]}/2, 1}}

```

(3) 방정식 풀기  
다음과 같은 연립미분방정식을 수치적으로 풀어서 그 해를 그림으로 나타낼 수 있다.

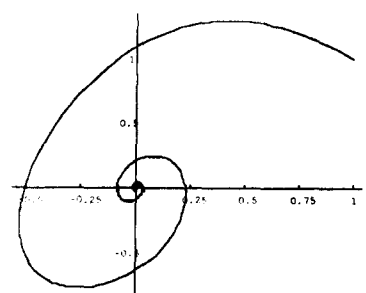
$$\begin{aligned} x'(t) &= -y(t) - x(t)^2 \\ y'(t) &= 3x(t) - y(t) \\ x(0) &= 1 \\ y(0) &= 1 \end{aligned}$$

위 연립미분방정식의 수치해를  $0 \leq t \leq 10$  구간에서 구하여  $x, y$  평면에 표시하면 아래와 같다.

```

sol = NDSolve[{x'(t) == -y(t) - x(t)^2,
 y'(t) == 3 x(t) - y(t), x(0) == 1,
 y(0) == 1}, {x, y}, {t, 10}]
{{x -> InterpolatingFunction[{{0., 10.}], <>],
 y -> InterpolatingFunction[{{0., 10.}], <>]}}
ParametricPlot[Evaluate[{x(t), y(t)} /. sol],
 {t, 0, 10}, PlotRange -> All]

```



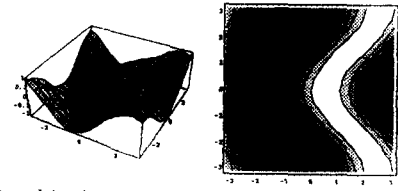
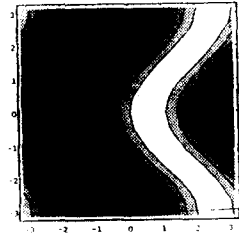
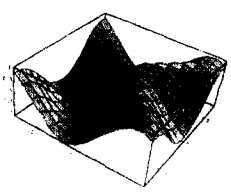
(4) 그래프  
CAS들은 대개 아주 유연한 함수 정의 기능과 막강한 그래픽 처리능력을 보유하고 있다.  
다음은  $\sin(x + \cos(y))$ 의 그래프를  $-\pi \leq x, y \leq \pi$

의 범위에서 다양하게 그려본 것이다.

```

sinCos = Plot3D[Sin(x + Cos[y]), {x, -Pi, Pi}, {y, -Pi, Pi}]
- SurfaceGraphics-
ensinCos = ContourPlot[Sin(x + Cos[y]), {x, -Pi, Pi}, {y, -Pi, Pi}]
- ContourGraphics-
Show[GraphicsArray[{sinCos, ensinCos}]]
- GraphicsArray-

```

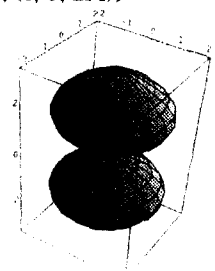


다음은  $f(z, t) = (2\sin(z)\sin(t), 2\sin(z)\cos(t), \cot(t))$ 로 표시되는 함수에 대한 3차원 매개변수 그래프이다.

```

ParametricPlot3D[{2 Sin[z] Sin[t], 2 Sin[z] Cos[t], z}, {z, -Pi, Pi}, {t, 0, 2Pi}]
- Graphics3D-

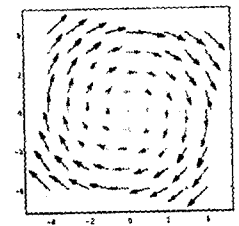
```



벡터장  $\vec{v}(x, y) = (y/4, -x/4)$ 을  $-4 \leq x, y \leq 4$  범위에서 표시해 보면 다음과 같다.

```

<<Graphics`PlotField` (* Mathematica 패키지
를 읽어들임 *)
data = Flatten(Table[{x, y}, {y/4, -x/4}],
{x, -4, 4, 1}, {y, -4, 4, 1}), 1];
ListPlotVectorField[data, ColorFunction-
>Hue, Frame->True]
    
```

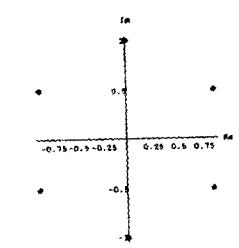


-Graphics-

다음의 예는  $x^6 = -1$ 을 풀고 그 해를 복소평면 상에 표시한 것이다.

```

ListPlot[
Map[{Re(#), Im(#)} &, x /. Solve(x^6 == -1)],
PlotStyle->PointSize(0.02),
AspectRatio->Automatic,
AxesLabel->{"Re", "Im"}]
    
```



-Graphics-

### 6. 맺음말

우리는 이제까지 CAS 및 symbolic calculation의 의미와 특징, 용도 및 CAS 중의 하나인 Mathematica의 실제 사용 예를 살펴보았다. 이 분야는 특히 우리나라에서는 아직까지 시작단계에 있지만 곧

누구나 하나의 CAS쯤은 능숙하게 다루어야 하는 시대가 곧 올 것이다. 그리고 이미 세계 각국에서는 공학 교육에 CAS를 많이 응용하고 있다. 다음 그림은 서울대 조선해양공학과 학생이 4학년 "선박설계 I" 과목의 설계 과제를 수행하면서 실적선의 선각모양을 간단히 CAS로 표현해 본 것이다. 여기서 강조하고 싶은 것은 다음과 같은 작업을 컴퓨터 그래픽에 대한 별다른 지식 없이도 약 20여줄의 명령어로서 수행할 수 있다는 것이다.



서울대 조선해양공학과에서는 1학년이 수강하는 "컴퓨터 개론 및 프로그래밍" 과목에서 symbolic calculation의 내용을 가르치고 있다[11]. 이 분야는 컴퓨터의 급격한 기능향상에 힘입어 큰 폭의 발전이 기대되는 새로운 분야이므로 많은 관심을 가져야 할 것이다.

### 참고 문헌

- [1] Beltzer, A.I., "Engineering analysis via symbolic computation - a breakthrough," Appl. Mech. Rev., Vol.43, pp.119-126, 1990.
- [2] Beltzer, A.I., A symbolic computation approach, Variational and finite element methods, Springer, Berlin, 1990.
- [3] Leff, L. and Yun, D.Y.Y., "The symbolic finite element analysis system," Comp. Struc., Vol.41, pp.227-231, 1991.
- [4] Chang, T.Y., Tan, H.Q., Zheng, D. and Yuan, M.W., "Application of symbolic method to hybrid/mixed finite elements and computer implementation," Comp. Struc., Vol.35,

- pp.293-299, 1990.
- [5] Ioakimidis, N.I., "Application of MATHEMATICA to the solution of elasticity problems by the finite element method," *Comp. Meth. Appl. Mech. Engng.*, Vol.102, pp.29-40, 1993.
- [6] Ioakimidis, N.I. and Anastasselou, E.G., "Gröbner bases in truss problems with MAPLE," *Comp. Struc.* Vol.52, pp.1093-1096, 1994.
- [7] Helzer, G., "Gröbner Bases," *The Mathematica Journal*, Vol.5, Issue 1, pp.67-73, 1995.
- [8] Patte, H. A., "Selecting computer mathematics," *Mechanical Engineering*, September, pp.82-84, 1995.
- [9] Wolfram, S., *Mathematica - A System for Doing Mathematics by Computer* 2nd Ed., Addison Wesley, Redwood City, California, 1991.
- [10] 강상균, 남기원, 전규창, *Mathematica?! - 입문에서 활용까지*, 성안당, 1993.
- [11] 신종계, *컴퓨터의 이해와 활용 <컴퓨터 개론 및 프로그래밍 교재>*, 서울대 조선해양공학과, 1995.

## \*    일   립   니   다   \*

1995년도 대한조선학회 추계연구발표회(개최지 : 충남대학교, 개최일자 : 95. 11. 9(목) ~ 10(금))에서 발표된 내용을 학회지 또는 논문집에 투고를 희망하시는 회원분께서는 다음 사항에 유의하시어 원고를 보내주시기 바랍니다.

### -   다   음   -

- ◆ 저자명과 소속명이 삭제된 심사용 원고 3부
- ◆ 국문요약 첨부
- ◆ 심사료 이만원 선납  
만원권 소액환 2장 동봉(이만원권 소액환은 피해 주시기 바랍니다)  
지로납부시에는 영수증 사본을 동봉(지로번호 : 7510658)
- ◆ 다음을 꼭 알려 주세요
  - 발표일자
  - 게재희망지(학회지, 국문논문집, 영문논문집 중 택일)
  - 논문게재시 게재료를 납부하실분
  - 별쇄본 신청 여부