

## 数式処理システム Maxima の利活用

文教大学情報学部情報システム学科准教授 佐久間拓也

### 1 はじめに

Maxima とは、数式処理システムのソフトウェアの一つです。しかも、OSS(オープンソースソフトウェア、数学資料 No.54「教材に役立つソフトウェア紹介」参照)であり、無償で入手が可能です。同様のソフトウェアで商用のソフトウェアには、Mathematica や Maple といったものがありますが、かなり高価なものです。

Maxima は、無償とはいえ商用のソフトウェアと比べるとユーザインターフェースなどで見劣りするところもありますが、機能は豊富でとても優れたソフトウェアです。数式の展開はあたりまえで、他にも因数分解、微分、積分、線形方程式、行列などもできます。もちろん 2D や 3D のプロットも可能です。

この Maxima の使い方および授業等での適用や応用を考えてみましょう。

#### 1.1 Maxima の入手とインストール

Maxima は、次の Web ページから入手できます。  
<http://maxima.sourceforge.net/>

Windows に対応した Maxima をダウンロードするには、まずこのページの左のメニュー「Download」そして「Sourceforge download page」をクリックします。そのページの「5.14.0-Windows」そして「maxima-5.14.0a.exe」をクリックすればダウンロードされます(2008 年 1 月現在)。あとはそのファイルを実行すれば、Maxima がインストールされます。なお利用のしやすさを考えると、wxMaxima か XMaxima はインストールしておきましょう(標準でインストールされます)。インストールされた Maxima を実行するには、メニューの「Maxima-5.14.0」から wxMaxima か XMaxima を選択してください。ここでは、wxMaxima を使うことにします。

#### 2 Maxima の使い方

wxMaxima を実行すると図 1 のようになり、ここの「INPUT」欄に命令を入力していきます(本来ならば行末に ; が必要なのですが

wxMaxima では省略できます)。命令を実行すると結果が表示されます。この時、入力した式が(%i1)のように%iのあとに番号がついたところに表示され、実行の結果が(%o1)のように%oのあとに表示されます。この(%in)、(%on)は記録され後でその式を利用することが出来るようになります。つまり入力を(%i10)+2\*(%o12)とすると、これは(%i10)のところを(%i10)の式に、(%o12)のところを(%o12)の式にするということになります。

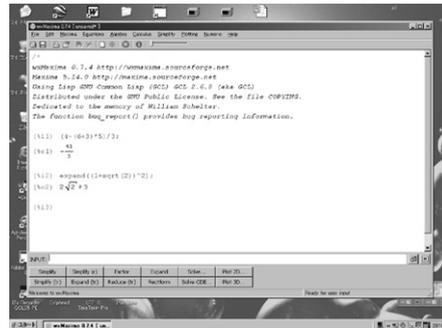


図 1 wxMaxima

#### 2.1 基本

加減乗除には、+、-、\*、/を使います。冪は^を、階乗は!を、平方根はsqrt()を使います。もちろん、変数はそのまま利用することが出来ます。なお文字(変数)の乗法の際には、\*を省略してはいけません。つまり、xy と x\*y は異なり、前者は xy という文字(変数)で、後者は x という文字(変数)と y という文字(変数)の乗法になります。

虚数単位 i は %i を、円周率 π は %pi を、e は %e を使います。絶対値を求めるには abs(), 複素数の絶対値を求めるには cabs() を使います。他に自然対数の log(), 三角関数の、sin(), cos(), tan(), csc(), sec(), cot(), 逆三角関数の asin(), acos(), atan(), acsc(), asec(), acot() などの関数があります。

さらに、リストと呼ばれる、複数のデータを順序で組にしたものを扱うことが出来ます。リストを作るには、コンマで区切り[]で括弧で囲みます。リストの要素には、数値だけではなく数式や別のリストなどもとることが出来ます。

文字(変数)に値や式を代入するには、:を使います。使用している間はずっと記憶しているので注意が必要になります。この関係を消すには kill() を使います。関数を定義する場合は、:= を使います。関数を消すときも kill() を使います。

```
変数名:変数の内容
関数名(変数名):=関数式
kill(変数名1,...
```

## 2.2 数と式

数式の展開には expand(), 因数分解には factor() を使います。factor は素因数分解もできます。

```
expand(多項式)
factor(多項式,変数名)
```

## 2.3 方程式

方程式を解くには、solve() を使います。一元一次方程式だけではなく、代数的に計算可能なものであれば連立方程式や  $n$  次方程式でもほぼすべて解いてくれます。2 つ以上の方程式がある場合は、方程式をリストで並べます。

```
solve(方程式,変数名)
```

## 2.4 関数とグラフ

グラフを書くには、2 次元は plot2d() を、3 次元は plot3d() を使います。

```
plot2d(関数式,[定義変数名,開始値,終了値])
plot3d(関数式,[x成分変数名,開始値,終了値]
,[y成分変数名,開始値,終了値])
```

## 2.5 その他

リストを、各要素が各要素番号を変数として作るときには makelist で作成できます。

```
makelist(計算式,カウンタ変数名,初期値,終了値)
```

なお、カウンタ変数は初期値から始まり終了値まで 1 ずつ増えながら変化します。

他にも、極限を計算するための limit() や微分するための diff() (多階微分・偏微分が可能です)、積分するときの integrate() (不定積分・定積分が可能です)、微分方程式を解くための desolve() または ode2(), 行列を扱うための matrix() など

があります。

## 3 Maxima の活用

今までは式の計算・展開・因数分解等を手作業でしていましたが、数式処理システムを使えば、それらはコンピュータがしてくれます。

コンピュータが計算するのでは、数学の学習に使えないと思われる方もいるでしょう。ですが、あえてコンピュータを利用し学習方法を変えて見るのも良いと考えます。たとえば、数学資料 No.55 「GRAPES の紹介と活用事例紹介」のようにグラフを観察して意味を考えさせるといったことが考えられます。ほかにも、自分の解答が正しいかどうかを確認するために利用するということにも使えます。

Maxima を使った場合には、どのようなものがあるか考えてみましょう。

### 3.1 式の展開と因数分解

式の展開をたくさん行わせて、そこからどのような法則性があるかを見つけ出させてみるというのはどうでしょう。たとえば、

```
(%i1) expand((2*x-3)*(x+5))
(%o1) 2x^2 + 7x - 15
```

のようなことをたくさん実行させて、そこから  $(ax+b)(cx+d) = acx^2 + (ad+bc)x + bd$  が発見出来るかを考えさせてみたり、

```
(%i1) expand((x+y)^2)
(%o1) x^2 + 2xy + y^2
(%i2) expand((x+y)^3)
(%o2) x^3 + 3x^2y + 3xy^2 + y^3
(%i3) expand((x+y)^4)
(%o3) x^4 + 4x^3y + 6x^2y^2 + 4xy^3 + y^4
:
```

を実行させて、どんな法則性があるか(たとえば  $n$  乗の時、 $x^{n-1}y$  の項の係数は  $n$  になるなど)考えさせてみる、といったことなどです。

また、展開や因数分解の結果を考えさせて、factor() や expand() で確認をしてみるようにする(この時、展開の問題の時に expand() を使って確認するだけではなく factor() でちゃんと因数分解出来て元の式に戻るかどうかを確認させる事も良いでしょう)、といったことがあります。

### 3.2 方程式とグラフの関係

Maxima では、異なる関数を同時にグラフで

表示することが可能です。そこで、2つの関数のグラフの交点がどのようにになっているかを考えさせて見ることができます。たとえば、 $y=2x-1$ と $y=x^3-2x^2+x+1$ の場合には、

```
(%i1) plot2d([2*x-1, x^3-2*x^2+x+1], [x, -4,4], [y, -10,10])
```

を実行して、描かれたグラフを観察して交点がどの辺りにあるかを考えさせてみます。そのときさらに、

```
(%i1) solve(2*x-1=x^3-2*x^2+x+1, x)
```

を実行して $2x-1=x^3-2x^2-x+2$ の解を求めてみて、グラフの交点と解との関係を考えさせてみると良いかと思えます。

### 3.3 パラメータを変化させてグラフに描く

makelist を使って、パラメータを変化させながら関数をリストで作成し、それをグラフで表示させるとおもしろいものが描かれることがあります。たとえば、 $y=2px-p^2$ で $p=-2, -1.8, -1.6, \dots, 1.8, 2$ と変化させたときのグラフがどうなるかを見るには、

```
(%i1) A: makelist(2*(-2+2/10*i)*x - (-2+2/10*i)^2, i, 0, 20)
(%i2) plot2d(A, [x, -4, 4], [y, -5, 10])
```

を実行すれば良いです。これを実行すると、図2のようになり $y=x^2$ が浮かび上がってきます。直線だけのグラフなのに曲線が見えてくるのは、生徒たちは不思議に感じるのではないのでしょうか？もちろんこのあとで、 $y=2px-p^2$ は $x=p$ における $y=x^2$ の接線であり、 $y=x^2$ は直線群 $y=2px-p^2$ の包絡線という」と、説明するのも良いでしょう。

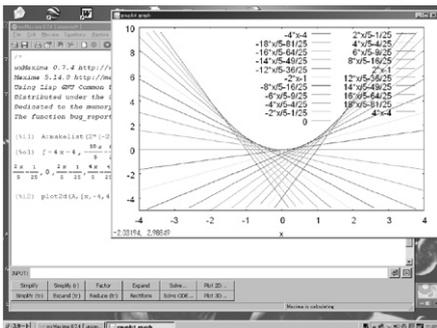


図2  $y=2px-p^2$ の直線群

### 3.4 3次元のグラフを描く

Maxima の活用として、なかなか手で描くことができない3次元のグラフを描くことも考えられます。3次元のグラフになるとかなり複雑なものもあり、そのようなものはMaximaなどを利用して描いた方がよいでしょう。例として、クラインの壺を描いてみます。なおグラフはマウスを利用して視点が変わりますので、いろいろな方向から見るすることができます。

```
(%i1) plot3d([5*cos(x)*(cos(x/2)*cos(y)+sin(x/2)*sin(2*y))+3.0)-10.0, -5*sin(x)*(cos(x/2)*cos(y)+sin(x/2)*sin(2*y))+3.0, 5*(-sin(x/2)*cos(y)+cos(x/2)*sin(2*y))], [x, -%pi, %pi], [y, -%pi, %pi], [grid, 40, 40])
```

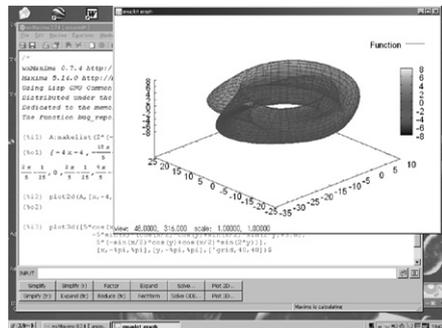


図3 クラインの壺

## 4 おわりに

Maxima は数式処理によって計算をする時にとっても役に立ちます。使い方によっては数学教育に役立てることも可能です。ですが、単なる計算をさせるのでは教育の本質を外れてしまいます。結果を見て、法則性を見いださせたり考察の補助にしたりと考えさせることがなければなりません。

Maxima を教育に使うとき、今までの教え方で利用を考えてしまうと単なる計算をさせるためのものになって意味がありません。まったく異なった方法で教えることを考えなければいけません。とてもおもしろい教育方法が考えられるのではないのでしょうか。ぜひ、皆さんもMaximaを利用して教育に活用してみてください。