

第17回グラフ電卓研究会（福井高専）、2016.06.18

スマホでMaximaを利用する

(元) 一関工業高等専門学校 梅野善雄

目次

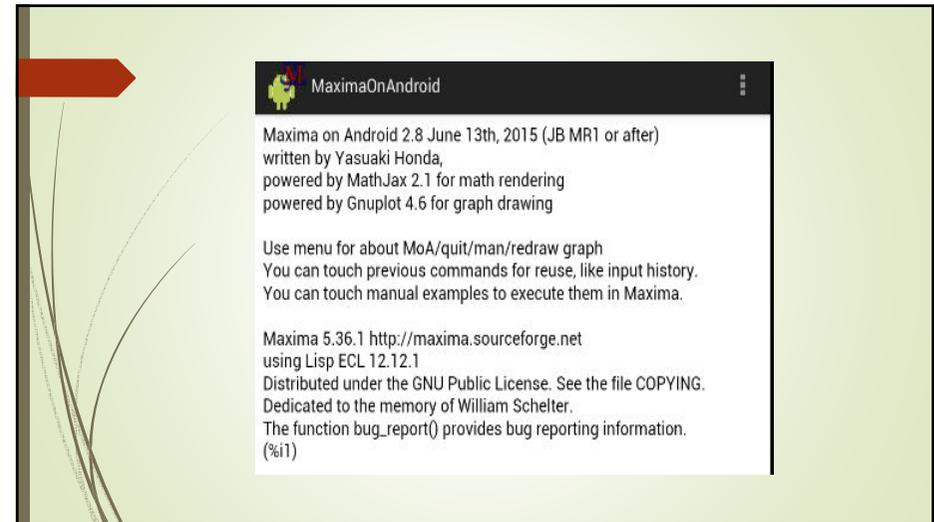
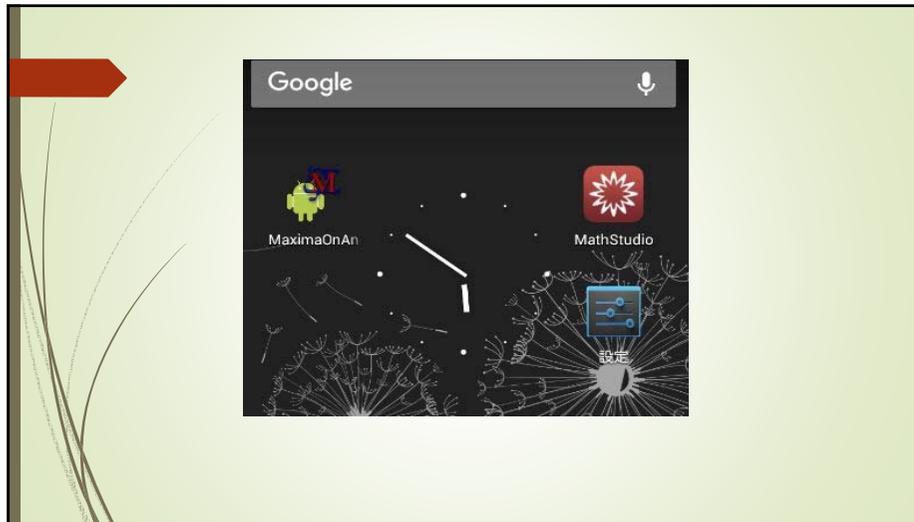
- はじめに
- Maximaのインストール
- 数と式の計算
- 方程式の解法
- 関数のグラフ
- 微分積分の計算
- 微分方程式の解法
- 線形代数の計算
- まとめ

はじめに

- Maxima (マキシマ)は、MathematicaやMapleよりも歴史の古い数式処理ソフト。
- 無料で利用でき、多様なOS版がある。
Unix, Windows, MacOS, Android
- Android版は、日本人の本田康晃氏によりリリースされた(2012年)。
- Android系のスマホやタブレットで、無料で、本格的な数式処理ツールを利用できることになる。
- 授業での利用は難しいかもしれないが、授業外学習で気軽に利用すれば、大きな教育効果が期待される。
- インストールの仕方と、主な利用の仕方について紹介する。

Maximaのインストール

- 「Google Plays」からダウンロードする。
MaximaOnAndroid で検索してインストールする。
- ファイルの大きさは約90MBで、60MBは外部メモリーに配置可能。
- Android版での数式表示は、MathJax が利用されている。
- グラフ表示は、Gnuplot による。(Maxima本体も同様)
- PC版では、各種機能をメニューから選択できるが、Android版では、そのような機能はない。
いちいちコマンドを打ち込む必要がある。
- Maxima の公式マニュアル(和文)が同包されている。
- マニュアルにある使用例を、簡単に参照する機能が追加されている。



数の計算

- 整数はメモリの許す範囲で表示できる。
- 小数は16桁で表示。
- 多倍長のときはbfloat を利用し、桁数はfpprec で指定する。
- 定数は、 π は %pi、ネイピア数 e は %e、虚数単位 i は %i
- 無限大は、 ∞ は inf、 $-\infty$ は minf、無限遠点は infinity
- 科学定数は全部で38個の定数を利用でき、値は2006 CODATA に依る。
- 文字への値の割り当ては、コロン : を利用する。a:3; など。
- 割り当ての解放は、kill. kill(a); など。
- 大文字と小文字は区別される。

式の計算

- 式の表示順は降べきの順になるが、アルファベットの最後の方から、小文字より大文字から表示。
- 式の展開はexpand、例：expand((x+1)^3)
- 整数域の因数分解はfactor、例：factor(x^3-1)
- 複素数域の因数分解はgfactor、例：gfactor(x^4+1)
- 通分はxthru、例：xthru(1/a+1/b)
- 式の整理はrat、例：rat(a*b+b*c+c*a)
- 簡略化はratsimp、例：ratsimp(式)
式の展開・因数分解・通分などをおこなって、できるだけ簡略化する。
- 変数の置き換えはratsubst、例：ratsubst(d, b+c, eq)
式eqの、b+cを、dで置き換える。
- 部分分数分解はpartfrac、例：partfrac(1/(x^3+1))

方程式の解法

- 多項式系の方程式、連立方程式は、solve で解が求められる。
例 : solve(a*x^2+b*x+c=0,x), solve([a*x+b*y=c,d*x+e*y=f], [x, y]);
- 3次方程式、4次方程式の解の公式が組み込まれている。
- 5次以上でも、整数域で因数分解できれば解ける。
- 区間(a, b) での実数解の個数を知るには、nroots。
例 : nroots(方程式, a, b)
- 実数解の近似値を有理数で求めるには、realroots。
例 : realroots(方程式)
誤差を指定することもできる。
- 多項式でない場合、区間(a,b)に実数解が1個だけであることが分かっているときその解の近似値を求めるには、find_root。
例 : find_root(方程式, a, b)

不等式の解法

- 不等式を解くには、パッケージをロードする。load(fourier_elim)
- 変数が1つだけの連立不等式の解を求めることができる。
例 : fourier_elim([不等式1, 不等式2, ...], [x])
- ただし、整数域で因数分解できる場合に限る。
- 実数域で因数分解した形で与えれば、解を求めることが可能。
- ただし、形式的な数式処理によるので、fourier_elim([(x+i)*(x-i)>0], [x]) では「max(-1, i)<x, x<min(-1, i)」が解として表示される。
- 未知数が2個の場合 :
例 : fourier_elim([x+y>1, x-y>1], [x,y]) → max(1-y, y+1)<x
例 : fourier_elim([x+y>1, x-y>1], [y, x]) → 1-x<y, y<x-1, 1<x

関数のグラフ (1変数)

- 1変数関数のグラフは、plot2d により描画される。描画は gnuplot による。
例 : plot2d(f(x), [x, a, b], [y, c, d]), plot2d([f(x), g(x)], [x, a, b])
- 軸の目盛りや曲線の色など、多彩なオプション指定ができる。
- 媒介変数表示 $x=f(t)$, $y=g(t)$ の場合は
例 : plot2d([parametric, f(t), g(t), [t, a, b]])
- $y=f(x)$ タイプと媒介変数タイプを同時描画させることができる。
例 : plot2d([parametric, f(t),g(t),[t,a,b]], h(x)), [x, c, d])
- 極座標 $r=f(\theta)$ は、媒介変数表示により描画できる。「r : f(θ)」としてから
例 : plot2d([parametric, r*cos(θ), r*sin(θ), [θ, a, b]])
- 陰関数 $f(x,y)=0$ のグラフは、パッケージ implicit_plot をロードする。
例 : implicit_plot(f(x,y)=0, [x, a, b], [y, c, d])

関数のグラフ (2変数)

- 2変数関数のグラフは、plot3d により描画される。複数の曲面も描画可能。
例 : plot3d(f(x,y), [x, a, b], [y, c, d]);
plot3d([f(x,y),[x,a,b],[y,c,d]], [g(x,y),[x,e,f],[y,g,h]])
- 曲面の色や、グリッド数など、多彩なオプション指定ができる。
- x軸中心に120度、z軸中心に30度回転した曲面が表示される。Android版では、表示画像を回転させることはできないが、表示する曲面の回転角を指定することはできる。
- 関数が定義されない点は指定区間の分割数を調整して描画可能。
- 等高線により平面に描画することもできる。
例 : contour_plot(f(x,y), [x, a, b], [y, c, d])
- 陰関数 $f(x,y,z)=0$ のグラフは、draw パッケージをロードすれば描画できる。
例 : draw(implicit(f(x,y,z)=0, x, a, b, y, c, d, z, e, f));

微分・積分の計算

- 極限値の計算は limit による。
例 : `limit(sin(x)/x, x, 0)`;
片側極限値、正負の ∞ の場合にも対応する。
- 導関数や偏導関数は diff により計算でき、微分回数を指定できる。
例 : `diff(f(x), x)`; `diff(f(x), x, 3)`; `diff(f(x,y), y)`;
- 微分する変数を指定しないと全微分が示される。
例 : `diff(f(x, y))` ; $\rightarrow d/dy f(x,y) dy + d/dx f(x,y) dx$
- 関数 $f(x)$ の指定された点でのテイラー展開は taylor による。
例 : `taylor(sin(x), x, 0, 10)`; $\leftarrow x=0$ で 10 次の項まで求める。
- 変数の従属関係を指定して陰関数の微分ができる。
例 : `depends(y, x)`; `diff(x^2+y^2=1, x)`;
 $\rightarrow 2y(d/dx y) + 2x = 0$
- 不定積分や定積分の計算は integrate による。
例 : `integrate(x*sin(x), x)`; `integrate(x*sin(x), x, 0, %pi)`;

微分方程式の解法

- 1階または2階の微分方程式の一般解は ode2 により求めることができる。
例 : `ode(微分方程式, y, x)` ;
- 微分方程式の指定は,
 $x^2y' + 3xy = \sin(x)/x$ であれば、`x^2*'diff(y,x)+3*x*y=sin(x)/x`
 $y'' + y(y')^3 = 0$ であれば、`'diff(y,x,2)+y*'diff(y,x)^3=0`
- 任意定数は、1階は %c、2階は %k1, %k2 で表される。
- 求められた解を % とするとき、初期条件を満たす特殊解を求めるには
1階で、 $y(\%pi) = 0$ であれば、`ic1(% , x=%pi, y=0)`;
2階で、 $y(0) = 0, y'(0) = 1$ であれば、`ic2(% , x=0, y=0, diff(y,x)=1)`;
(注 : 「'」を付けなくてもかまわない)
- 求められた解を % とするとき、境界条件から特殊解を求めるには
 $y(0) = 1, y(1) = 3$ であれば、`bc2(% , x=0, y=1, x=1, y=3)`;
- `desolve` は、ラプラス変換を利用して解を求める。ode2 の方が良い。

線形代数の計算

- 行列は、matrix により定義する。
例 : `A:=matrix([1, 2], [3, -4])` ;
- 行列式は `determinant(A)`
- 階数は `rank(A)`
- 固有多項式は `charpoly(A, x)` $\rightarrow x^2+3x-10$
- 固有値は `eigenvalues(A)` $\rightarrow [[-5, 2], [1, 1]]$
(注) 後半の $[1, 1]$ は重複度を表す。
- 固有ベクトルは `eigenvectors(A)`
 $\rightarrow [[[-5, 2], [1, 1]], [[1, -3]], [[1, 1/2]]]$
(注) 最初に固有値と重複度が示される。
- ガウスの消去法で対角成分を 1 にするには `echelon(A)`
(注) 対角成分が「1」の上三角行列に変形されるだけ。

まとめ

- 学生の半数近くが所有する Android 系スマホに、無料で本格的な数式処理ソフトをインストールすることができる。
- PC 版にも無料でインストールでき、同じコマンドで操作できる。
- 結果の表示ツールとしてばかりではなく、紙と鉛筆替わりに利用することで、手計算を行うことなく数学的な思考展開を行うことができる。
- 多様なグラフを簡単に確認できることは、学生の理解度アップに大きく寄与するのではないかとと思われる。
- 日常的に身につけているスマホで「Maxima」という本格的な数式処理を利用することができる。それを使いこなせるようになれば、大きな教育効果が期待される。