

数式処理システム Maxima の Linux システムへの導入

岡本 将彦*

神戸大学大学院 総合人間科学研究科

高橋 正†

神戸大学 発達科学部

1 Maxima の特徴

“Maxima” ([1]) とは、MIT の DOE (Department of Energy) で開発されていた “Macysma” を、William F. Schelter 氏が Common Lisp で実装したものが土台となっている。この Macysma は 1960 年代から開発されており、そのバリエーションの一つとして 1980 年代より Maxima は開発された。そして、近年 Schelter 氏が DOE より許可を得て、GPL ライセンスの下で配布されるようになったという経緯を持つ。

GPL ライセンスで配布されている事は、Maxima の大きな特徴である。この GPL ライセンスに則って、Maxima は無料で利用する事が可能であり、オープンソースであるため、Schelter 氏が亡くなられてしまった現在でも、多数の開発者によって開発が進められている ([2])。

現在の最新バージョンは 2001 年 5 月に発表された「v.5.6」である。最新バージョンでは、簡易的な Web Browser を備えた GUI が実装されており、商用の数式処理システムに迫る機能を有しつつあるといえる。

2 Maxima の入手先

Maxima をインストールするには、tcl/tk 及び wish が必要とされる。これらが利用できない場合は、先にパッケージをインストールしなければならない。

Maxima は Common Lisp で実装されている為、ソースパッケージをコンパイルするには、gcl や clisp 等の Common Lisp 処理系が必要であり、一般的には gcl が利用されるようである。Maxima 及び gcl のソースパッケージは、Shelter 氏の Web page ([1]) よりダウンロードする事が可能であり、clisp のソースパッケージは clisp の Web page ([3]) よりダウンロードする事が可能である。

*masahiko@boogie.h.kobe-u.ac.jp

†takahasi@kobe-u.ac.jp

また、Maxima の rpm 形式のバイナリパッケージが、symaxx の Project Page ([4]) よりダウンロードする事が可能である。

Maxima の Project Page ([2]) にはソースやバイナリのパッケージは配布されていないが、開発版の Maxima が cvs によって管理されている。

3 Maxima のインストール

Linux はディストリビューションによって、パッケージ管理方法が大別される。今回私は、パッケージ管理方法の異なるいくつかの Linux 上において Maxima のコンパイルを試みた。

3.1 Kondara MNU/Linux へのインストール

Kondara MNU/Linux (以下 Kondara) では rpm 形式でパッケージ管理を行なう。今回、OS のバージョンは 2.0 及び 2.1 を用いた。

ソースからコンパイルしてインストールする場合、Kondara では gcl のコンパイルする段階でエラーが生じる。以下は、エラーメッセージの一部分である。

```
(cd unixport; make saved_gcl)
make[1]: Entering directory '/home/masahiko/source/gcl-2.4.0/unixport'
cc -c -I../h -c -o ../mp/mpi.o ../mp/mpi.c
cc -c -I../h -c -o ../mp/mp2.o ../mp/mp2.c
make[1]: *** No rule to make target '../mp/libmport.a', needed by
'raw_gcl'.
Stop.
make[1]: Leaving directory '/home/masahiko/source/gcl-2.4.0/unixport'
make: *** [all] Error 2
```

同様のエラーが Red Hat Linux 7.* 上でも発生する事が確認されているが、解決策は見つかっていないようである。私も、エラーを回避するよう試行錯誤を行なったが解決は出来なかった。ちなみに、Mandrake Linux ではコンパイルに成功するようであるため、Red Hat 系の Linux に生じる問題ではないようである。

なお、バイナリパッケージからインストールする場合は、特に問題なくインストールでき、利用する事ができた。

3.2 Debian GNU/Linux へのインストール

Debian GNU/Linux (以下 Debian) では deb 形式でパッケージ管理を行なう。今回、OS のバージョンは 3.0 “woody”(testing) を用いた。

Debian では Maxima は Official Package として用意されている為、apt を用いる事で依存関係を解決してインストールする事ができる。実際に行なったコマンドは、以下の一行である。

```
# apt-get install maxima
```

また、Debian で用意されているパッケージには、maxima を利用する手助けとなる Emacs 用の Lisp がいくつか含まれており、自動でコンパイルされるようになっている。

3.3 Plamo Linux でのコンパイル

Plamo Linux は Slackware を元に、日本語版のソフトウェアをパッケージングした、ディストリビューションである。

今回 Plamo Linux で、gcl と maxima のソースパッケージのコンパイルが可能なのは確認したが、インストール、プログラムの動作は確認してない。

4 Maxima の使用方法

Maxima は、コマンド "maxima" によってコンソール上で起動させることができるが、X-window が起動しているならばコマンド "xmaxima" によって、図 1 のような GUI を起動させることができる。

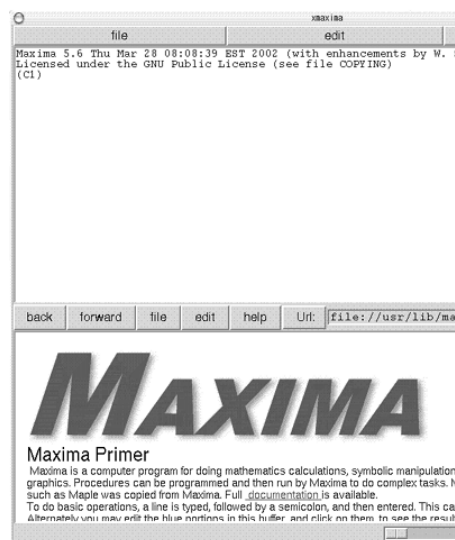


図 1: XMaxima の起動

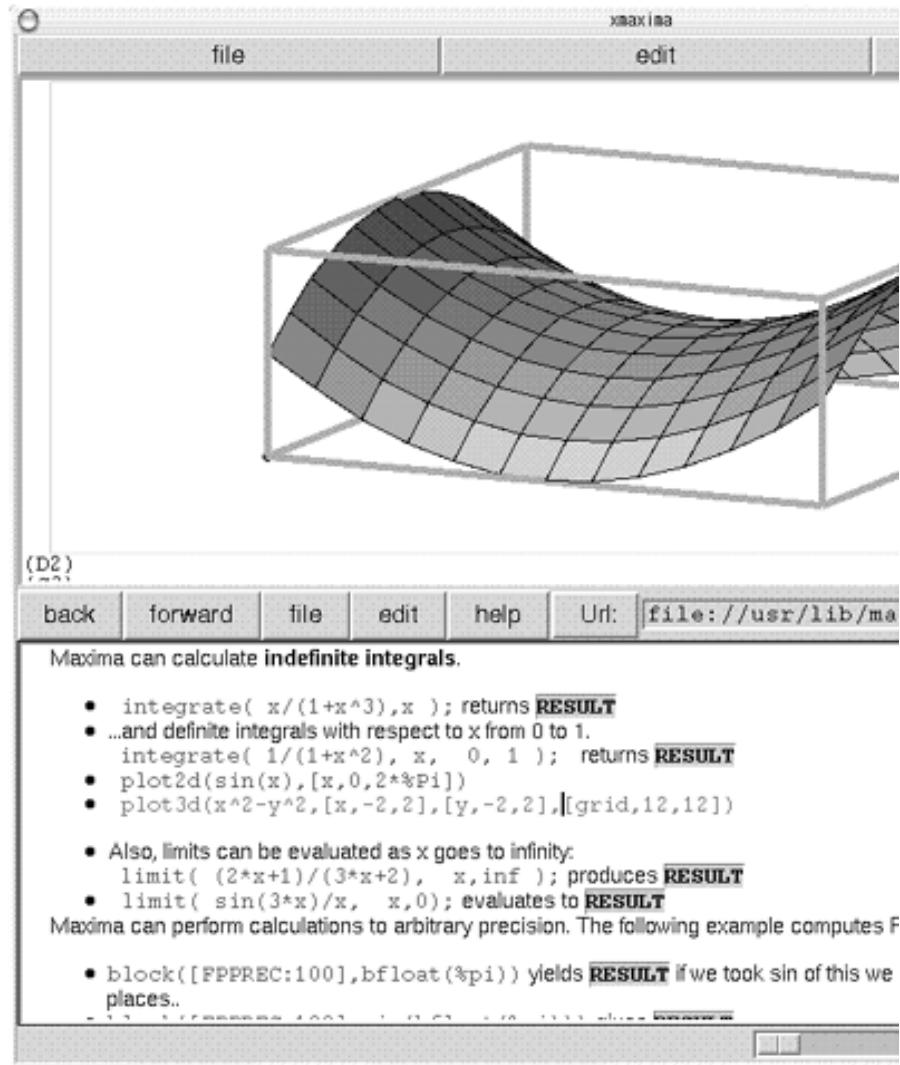


図 2: XMaxima による 3D 表示

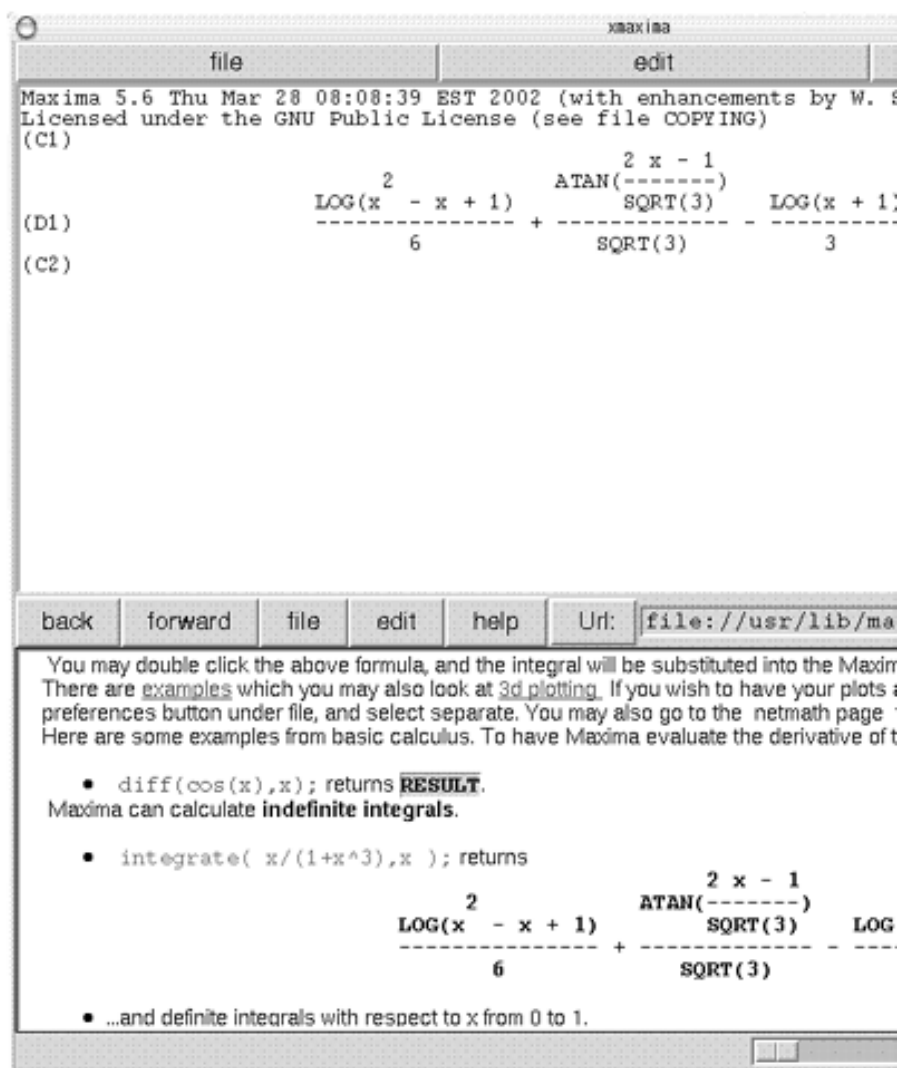


図 3: ブラウザを使った計算

この xmaxima のウィンドウは上下に二分割されており、上は通常のコンソールであるが、下はブラウザとなっており、下のブラウザに表示されている式をダブルクリックして、上のコンソールで結果を表示したり、ブラウザにインタラクティブに結果を表示することができる (図 2 ならびに図 3 参照)。

また、Emacsen 用の Lisp をインストールした場合は、Emacsen から「M-x run-maxima」とコマンドを実行することで、Emacsen 上で maxima を起動させることも可能である。

パッケージには膨大な英語のマニュアルが付属しており、また、ぼんぼこ様によって Web 上でマニュアルの日本語訳が公開されている。詳しい使用方はそちらのマニュアルを参照していただきたい。

参 考 文 献

- [1] Original website for Maxima, <http://www.ma.utexas.edu/maxima.html>
- [2] Maxima at SourceForge, <http://maxima.sourceforge.net/>
- [3] Clisp website, <http://clisp.cons.org/>
- [4] Symaxx project page (at SourceForge), <http://sourceforge.net/projects/symaxx/>