Aplicación de XMaxima a la docencia de las matemáticas

Francisco Palacios*

Dep. Matemática Aplicada III
Escuela Universitaria Politécnica de Manresa
Universidad Politécnica de Cataluña

Mayo, 2004.

Resumen

XMaxima es un programa de cálculo simbólico bajo licencia GNU-GPL. La versión actual (Maxima 5.9.0, publicada en febrero de 2003), posee distribuciones para Linux, Unix y MS-Windows, presenta un entorno de trabajo amigable y dispone de unos recursos de cálculo simbólico, numérico y gráfico que lo hacen especialmente apto como herramienta docente en asignaturas de contenido matemático, científico o técnico. Esta comunicación persigue un doble objetivo: (1) Analizar las ventajas e inconvenientes del uso de software libre como herramienta docente para asignaturas de contenido matemático en los primeros cursos de licenciaturas de ciencias e ingenierias. (2) Presentar brevemente las principales características de XMaxima que son relevantes para su uso como herramienta docente.

1 Introducción

Un Sistema de Computación Algebraica (CAS) es un programa informático que nos permite realizar manipulaciones algebraicas como, por ejemplo, operar con números en forma exacta, manejar expresiones con variables, factorizar enteros o polinomios, resolver ecuaciones de forma exacta, calcular derivadas y primitivas, operar con matrices y calcular determinantes que contienen parámetros, etc. También permite el empleo de enteros de longitud arbitraria y realizar cálculos aproximados con un número arbitrario de dígitos.
Entre los programas comerciales de computación algebraica hay que destacar Mathematica y Maple que, hoy en día, se han convertido en una herramienta usual en ambientes científicos y académicos. Estos programas permiten,

*e-mail: `francisco.palacios@upc.es`

además de la manipulación algebraica, la realización de excelentes representaciones gráficas y poseen un lenguaje de programación de alto nivel que admite incorporar en los programas los recursos gráficos y del CAS. Otro hecho destacable es que, en los últimos años, los CAS han rebasado el marco de los ordenadores personales y nos encontramos con que algunas calculadoras de bolsillo de gama media–alta incorporan un CAS de prestaciones realmente notables.
El uso de herramientas informáticas en la docencia de las matemáticas nos permite un enfoque más experimental del proceso de aprendizaje, facilitando que el alumno explore distintas posibilidades mediante la realización de cálculos, gráficos o desarrollos algebraicos que, manualmente, serían inabordables. También permite que el alumno realice un mayor número de ejercicios, combinando la resolución manual e informática de tareas repetitivas, tales como la resolución de sistemas de ecuaciones lineales o el cálculo de determinantes. Además, en algunas disciplinas como el Cálculo Numérico, la aparición de esquemas iterativos de cálculo exige que, para una buena comprensión de la materia, el alumno deba realizar algunos programas elementales. Por otra parte, parece razonable que uno de los objetivos de la formación matemática del futuro científico o ingeniero sea, precisamente, la adquisición de una cierta experiencia en el manejo de software matemático avanzado.
Los contenidos matemáticos que se imparten en la etapa inicial de formación universitaria pueden variar considerablemente según el tipo de licenciatura o ingeniería, aunque, a grandes rasgos, podríamos establecer como requisitos básicos que nuestro software matemático de apoyo a la docencia sea capaz de:

- Representar funciones de una variable.
- Manipular polinomios y funciones racionales (descomposición en fracciones simples).
- Resolver ecuaciones, en modo exacto y aproximado.
- Calcular derivadas y primitivas de forma simbólica y calcular aproximaciones numéricas de integrales definidas.
- Resolver ecuaciones diferenciales ordinarias en forma simbólica y aproximada.
- Representar funciones de dos variables.
- Realizar operaciones matriciales, a ser posible, de forma simbólica.

También es importante que disponga de un lenguaje de programación de alto nivel que permita incorporar los recurso algebraicos y gráficos; esto es,

que nos permita, por ejemplo, realizar derivadas formales y definir funciones en tiempo de ejecución.
Además, teniendo en cuenta la limitación de tiempo y de recursos, tanto por parte de profesores y alumnos como de las instituciones universitarias, sería deseable que, junto a las anteriores características de índole matemático, nuestro software docente pudiera satisfacer otro tipo de requisitos, tales como:

- Disponibilidad de licencias: posibilidad de distribución legal a los alumnos y, en general, a todas las personas interesadas en nuestros cursos.
- Facilidad de uso para usuarios inexpertos: entorno de trabajo amigable y lo más sencillo posible.
- Facilidad de instalación.
- Independencia del sistema operativo: capacidad para funcionar bajo MS-Windows 95/98/Me/2000/XP y bajo Linux.
- Bajo consumo de recursos: poca ocupación de disco y posibilidad de funcionar con procesadores anticuados.
- Estabilidad de versiones: mantenimiento de las características básicas durante el mayor tiempo posible, de forma que los manuales y materiales docentes tengan un tiempo razonable de vigencia.

Si atendemos únicamente al primer grupo de características planteado, está claro que programas como Maple o Mathematica aparecen como una excelente opción; sin embargo, si consideramos el segundo grupo de características, vemos que la idoneidad de estos programas como herramientas docentes en los primeros cursos universitarios es, cuanto menos, discutible.
Una posible alternativa que, a mi juicio, se ajusta casi perfectamente a las exigencias planteadas, es la versión de Maxima distribuida bajo licencia GNU–GPL.

2 El programa Maxima

2.1 Un poco de historia

A principios de la década de 1960, se creó en el MIT (Massachusetts Institute of Technology) una unidad de investigación denominada Proyecto MAC (Mathematics and Computation). El objetivo del Proyecto MAC era construir herramientas computacionales inteligentes que ayudaran a científicos e ingenieros en la comprensión de los modelos matemáticos de los procesos físicos. El proyecto estaba financiado principalmente por el Departamento

de Defensa y el Departamento de la Energía (DOE) de los EE.UU. y por la agencia ARPA (Advanced Research Projects Agency). El Proyecto MAC fue el precursor del actual LCS (Laboratory for Computer Science) del MIT, que ha jugado un papel decisivo en el desarrollo de herramientas tales como la hoja de cálculo, la red local Ethernet o la misma Internet.
A finales de la década de los 60, empezó a desarrollarse el Proyecto Macsyma (MAC's SYmbolic MAnipulator), cuyo objetivo era la automatización del tipo de operaciones que realizan los matemáticos, como un primer paso hacia la comprensión de la capacidad de los ordenadores para desarrollar un comportamiento inteligente. Macsyma fue el primer programa de cálculo simbólico, a partir de él surgieron una larga serie de variantes: VMaxima, Symbolics Maxima, DOE-Macsyma, Paramacs, incluso puede considerarse que Macsyma es precursor de Maple y Mathematica. Maxima es una variante de DOE-Macsyma mantenida desde 1982 hasta 2001 por William Shelter en la Universidad de Texas. En 1998, se obtuvo la autorización del DOE para la distribución de Maxima bajo licencia GNU-GPL.

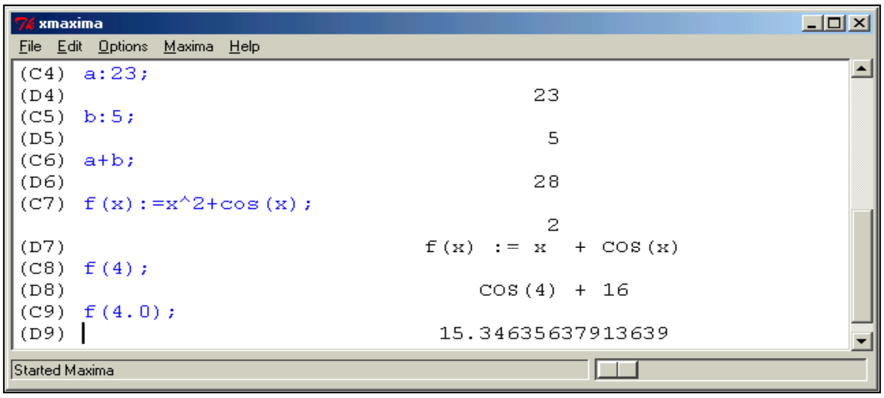
2.2 XMaxima

XMaxima es una implementación de Maxima basada en TCL/TK que puede ejecutarse en entornos Unix, Linux y MS-Windows y que está bajo licencia GNU-GPL.
La versión actual, Maxima 5.9.0, se publicó en febrero de 2003. Aparte del código fuente, existen .rpms para la instalación en Redhat y .deb para la instalación en Debian. La instalación en MS-Windows 95/98/Me/XP se lleva a cabo mediante un instalador de 10Mb, que funciona sin problemas; una vez instalado, el programa ocupa aproximadamente 28Mb. Los programas de instalación y la documentación pueden obtenerse en la página oficial de Maxima: `http://maxima.sourceforge.net`

2.3 Funcionamiento básico

En el entorno de trabajo se combina una ventana de tipo consola y el navegador NetMath, que permite visualizar y editar documentos HTML que interactuan con Maxima.
La consola permite desarrollar una sesión de trabajo interactivo: recoge las entradas y presenta los resultados. El funcionamiento es similar al de la consola de Matlab o Octave, aunque en la consola de XMaxima se puede ejecutar las ordenes entradas en líneas anteriores, desplazándose sobre ellas y pulsando `Enter`.
Podemos crear un fichero de texto con una copia de la sesión de trabajo mediante la opción `Save Console to File` del menú `Edit`. También se pueden ejecutar lotes de comandos contenidos en ficheros de texto con las opciones `Batch File` y `Batch File Silently` del menú `File`.

La líneas de órdenes deben terminar en punto y coma (como en Maple); si terminan en `$`, se omite la presentación de resultados. La asignación de valores a variables se hace mediante dos puntos `(:)`; para definir funciones se usa dos puntos seguidos de un igual `(:=)`. La siguiente ventana muestra la asignación de valores a las variables `a`, `b` y la definición de la función `f(x)` y su evaluación para un valor, en forma exacta y decimal. Las líneas con etiquetas `(C4)`, `(C5)`, ..., son entradas y las líneas con etiquetas `(D4)`, `(D5)`, ..., son las respuestas de Maxima. El uso de mayúsculas o minúsculas en las entradas no es relevante.



2.4 Documentación

En la página principal de Maxima, podemos encontrar el Manual de Maxima [4], un manual de referencia de DOE-Maxima [1] y un tercer manual aún incompleto: The Maxima Book [3]. Como es habitual en el software libre, una de las principales dificultades es la falta de una documentación adecuada. Los manuales de Maxima y DOE-Maxima son completas referencias técnicas, pero apenas tienen ejemplos; además, no se refieren a la actual versión y algunas de las características descritas no funcionan correctamente. En cuanto a The Maxima Book, en este caso el enfoque es más didáctico, con abundantes ejemplos, pero está aún en fase de elaboración.
Aparte de estas carencias, todas la obras citadas están en inglés y, por lo tanto, no son un material docente adecuado. Parece claro que si pretendemos emplear Maxima como herramienta docente, va a ser imprescindible elaborar algunos materiales docentes sobre el funcionamiento general de Maxima y sobre algunos ejemplos de aplicación.

3 Algunos ejemplos

Esta sección contiene algunos ejemplos elementales de las posibilidades de Maxima. Recordemos que las líneas con etiquetas `(C1)`, `(C2)`, ..., corresponden a las entradas del usuario; las líneas con etiquetas `(D1)`, `D(2)`, ..., presentan las respuestas de Maxima.

3.1 Funciones

Definición de una función de dos variables.

```
(C1) f(x,y):=x^2+y^2+cos(x*y);
                             2   2
(D1)          f(x, y) := x + y + COS(x y)
(C2) f(2,3);
(D2)                 COS(6) + 13
(C3) f(a,b);
(D3)             COS(a b) + b + a
```

Definición de una función vectorial.

```
(C4) g(t):=matrix([t,t^2,t^3]);
                                        2   3
(D4)           g(t) := MATRIX([t, t , t ])
(C5) g(1/2);
                        [ 1 1 1 ]
(D5)                    [ - - - ]
                        [ 2 4 8 ]
```

3.2 Límites

Las opciones: `plus`, `minus`, indican límites laterales por la derecha y por la izquierda; `inf` designa $+\infty$ y `minf` representa $-\infty$.

Límites con parámetros.

```
(C6) limit(sin(a*x)/x,x,0);
(D6)                         a
```

Límites en infinito.

```
(C7) limit(x-sqrt(x^2-x),x,inf);
                        1
(D7)                    -
                        2
(C7) limit(x-sqrt(x^2-x),x,minf);
(D7)                   MINF
```

Límites laterales.

```
(C8) limit(x*exp(1/x),x,0,plus);
(D8)                    INF
(C9) limit(x*exp(1/x),x,0,minus);
(D9)                      0
```

3.3 Derivación

La orden `diff(f(x),x,n)` calcula la derivada n-ésima de `f(x)`. El número e se representa por `%E`; análogamente, el número π se representa por `%pi`.

```
(C10) diff(x^2*exp(x),x);

                    2   x        x
(D10)              x  %E + 2 x %E

(C11) diff(x^2*exp(x),x,4);

                 2   x        x       x
(D11)           x  %E + 8 x %E + 12 %E

(C12) diff(2*x^2*y+2*x*y,x);
(D12)               4 x y + 2 y
(C13) diff(x^2*y*2+2*x*y,y);
                            2
(D13)               2 x + 2 x
```

3.4 Integración

Para calcular primitivas, usamos `integrate(f(x),x)`. Si especificamos los límites de integración, se calcula la integral definida simbólicamente; para obtener una aproximación decimal de una expresión exacta, podemos usar `ev(expresión,numer)`.

```
(C14) integrate(x^2*cos(x),x);

              2
(D14)      (x  - 2) SIN(x) + 2 x COS(x)

(C15) integrate(1/sin(x),x,1,%Pi/2);

           LOG(COS(1) + 1)     LOG(1 - COS(1))
(D15)      ---------------  -  ---------------
                  2                   2

(C16) ev(D15,numer)

(D16)               0.60458244594159
```

Los límites de integración pueden ser infinitos.

```
(C17) integrate(exp(-x^2),x,0,inf);
```

```
                              SQRT(%PI)
(D17)                         ---------
                                  2
```

También podemos incluir parámetros en los integrandos, en ese caso, el comando `assume` permite asignar atributos a los parámetros.

```
(C18) assume(a>0);
(D18)                          [a > 0]
(C19) integrate(exp(-a*x^2),x,0,inf);

                              SQRT(%PI)
(D19)                         ---------
                              2 SQRT(a)
```

Si Maxima no puede calcular una primitiva, entonces es incapaz de determinar el valor de la integral definida.

```
(C20) integrate(exp(x^3)*cos(x),x,0,1);
                            1
                           /     3
                           [    x
(D20)                      I  %E  COS(x) dx
                           ]
                           /
                            0
```

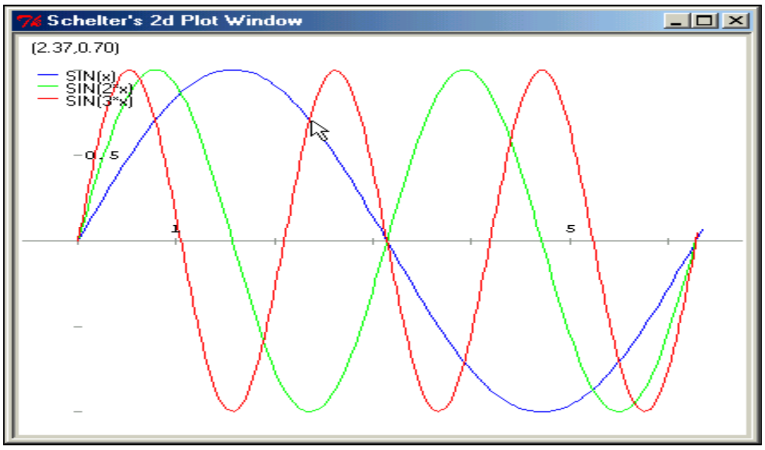
En ese caso, el comando `romberg` proporciona una aproximación numérica. Para usar `romberg`, debemos definir el integrando como función.

```
(C21) f(x):=exp(x^3)*cos(x);

                                 3
(D21)        f(x) := EXP(x ) COS(x)

(C22) romberg(f(x),x,0,1);
(D22)              1.071172734201934
```

3.5 Gráficos

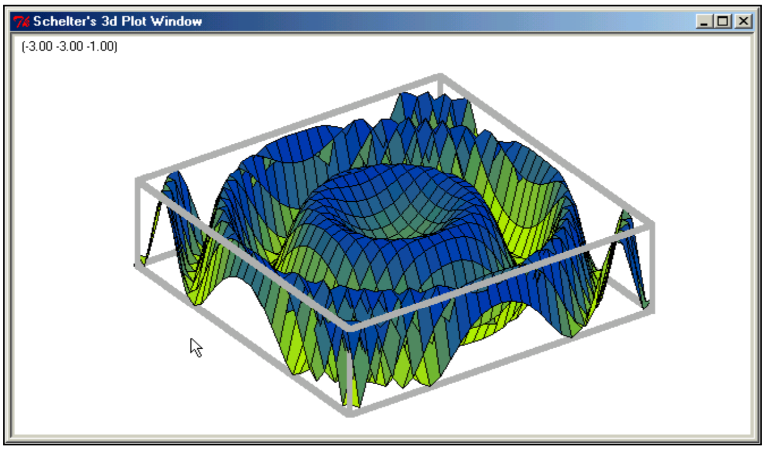
Maxima puede representar simultáneamente varias funciones de una variable. Señalando con el cursor en la ventana gráfica, se obtienen las coordenadas de los puntos.

```
(C23) plot2d([sin(x),sin(2*x),sin(3*x)],[x,0,2*%pi]);
```



También produce excelentes gráficos tridimensionales, que pueden girarse con el ratón.

```
(C28) plot3d(sin(x^2+y^2),[x,-3,3],[y,-3,3]);
```



3.5.1 Programación

Maxima posee un completo lenguaje de programación. El siguiente programa:

```
f(x):=exp(x)-2+x^2;
xx[0]:1.0;
define(df(x),diff(f(x),x))$
for i:1 thru 5 do (
     xx[i]:ev(xx[i-1]-f(xx[i-1])/df(xx[i-1])),
     display(xx[i])
)$
```

nos permite calcular 5 iteraciones de método de Newton-Raphson para la ecuación $e^x - 2 - x^2 = 0$, a partir del valor inicial $x_0 = 1$; obsérvese que la función derivada se define en tiempo de ejecución. El programa puede guardarse en un fichero de texto y ejecutarse con la opción `Batch File` del menú `File`. El ejecutar el programa, se produce la salida:

```
xx = 0.63582467285126
  1
xx = 0.54315669269868
  2
xx = 0.53729735863418
  3
xx = 0.53727444952344
  4
xx = 0.53727444917386
  5
```

4 Conclusiones

Los Sistemas de Computación Algebraica constituyen una buena herramienta para el aprendizaje y la docencia de las matemáticas y su uso debiera generalizarse. Existen programas comerciales, como Mathematica o Maple, que son muy adecuados para el uso profesional, pero que tienen agunos inconvenientes importantes, como su alto costo económico y la gran exigencia de recursos de hardware. El programa XMaxima, distribuido bajo licencia GNU-GPL, permite cubrir sobradamente las necesidades matemáticas que aparecen en los primeros cursos de las licenciaturas de ciencias e de ingeniería y, si se elabora una documentación docente adecuada, puede constituir una excelente alternativa a los programas comerciales.

Referencias

[1] Clarson, Michael. *DOE-Maxima Reference Manual. Ver. 5.9*, 2002. `http://maxima.sourceforge.net`.

[2] NetMath. Universidad de Texas. `http://www.ma.utexas.edu/users/wfs/netmath/netmath.html`

[3] Ney de Souza, Pablo; Fateman, R.J.; Moses, J.; Yapp, C. *The Maxima Book.* 2003. `http://maxima.sourceforge.net`.

[4] Shelter, William. *Maxima Manual.* `http://maxima.sourceforge.net`.