Scilab Datasheet

Computation in Scilab

Scilab is a programming language associated with a rich collection of numerical algorithms covering many aspects of scientific computation problems. From software point of view, Scilab is an interpreted language. This generally allows to get faster development processes, because the user directly accesses a high-level language, with a rich set of features provided by the library.

By design, Scilab was created to be able to perform matrix operations as fast as possible. The building block for this feature is that Scilab matrices are stored in an internal data structure which can be managed at the interpreter level. Most basic linear algebra operations, such as addition, substraction, transpose or dot product are performed by a compiled, optimized, source code. Performance is achieved through vectorization, using high-level linear algebra features.

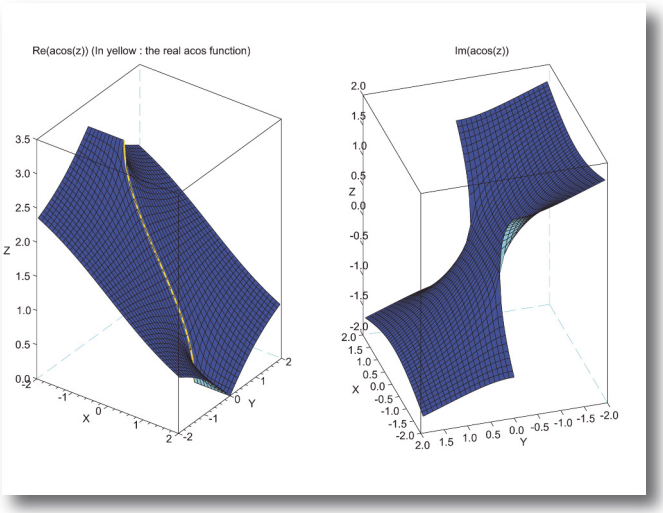

Figure 1: The acosh function in the complex plane. The acosh function is discontinuous across the line y=0 for x<=0 (branch cut).

Features

- Matrix algebra: +, -, *, ./ , ^, etc.
- Typical statement is C=A*B where A, B are real or complex matrices of doubles.
- More that 60 elementary functions: cos, sin, exp, log, ...
- Typical statement is y=sin(x) where x is real or complex matrix.
- Matrix functions: expm, pow, logm, cosm, etc.
- Special functions: gamma, beta, erf, etc.
- Dense high level linear algebra: backslash, lu, spec, svd, chol, least squares.
- Typical statement is x=A\b where A,b are real or complex matrix of doubles.
- Floating point number functions: %eps, frexp, number_properties, ieee, nearfloat.
- Test matrices: testmatrix (Franck, Hilbert, Magic squares), Toeplitz.
- Discrete maths: factorial, gamma, gammaln, binomial, permute, perms, ...

The following example is a typical Scilab session illustrating the main purpose of Scilab. We create a 2000-by-2000 matrix of doubles with Normal entries, then solve the associated linear equation and compute the forward error.

```
-->n=2000;
-->A=grand(n,n,"nor",0,1);
-->b=ones(n,1);
-->x=A\b;
-->norm(A*x-b)
 ans  =
    2.776D-15
```

Open-Source Libraries

- BLAS: routines that provide standard building blocks for performing basic vector and matrix operations.

http://www.netlib.org/blas/

- LAPACK: routines for solving systems of simultaneous linear equations, least-squares solutions of linear systems of equations, eigenvalue problems, and singular value problems.

http://www.netlib.org/lapack/

- SLATEC (e.g. Bessel): a comprehensive software library containing over 1400 general purpose mathematical and statistical routines written in Fortran 77.

http://www.netlib.org/slatec/

- SPECFUN (e.g. calerf): a collection of transportable Fortran programs for special functions.

http://www.netlib.org/specfun/

- Macros (e.g. erfinv).
- Fortran (e.g. complex functions such as complex tan).

Why you should use Scilab for numerical computing

- Matrix language: simple, fast if vectorized statements are used.
- Optimized BLAS optimized implementations (using Intel MKL under Windows and ATLAS under Linux).
- Scilab can be faster than what a typical user could easily make, even in a compiled language such as C or Fortran: vectorization provides the performance, at the library level.
- Rich set of elementary functions: e.g. sind (degree), sinh (hyperbolic), sinm (matrix), sinhm (hyperbolic matrix), etc.

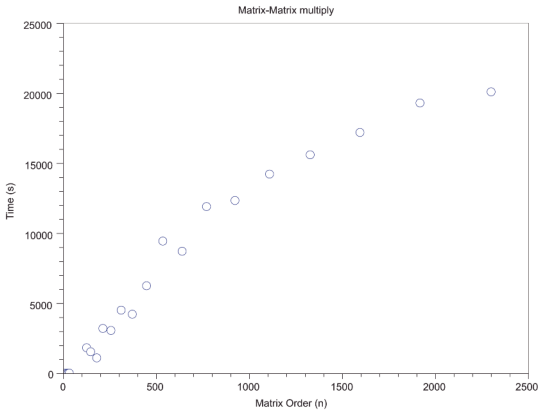

Figure 2: Performance of the Matrix-Matrix multiplication on a Windows 32 bits system with Scilab 5.3, Intel Xeon E5410 4*2.33 GHz, and the Intel MKL

Classical methods in numerical computing

- Differential equations and Integration,
- Nonlinear equations,
- Optimization,
- Probabilities / Statistics,
- Interpolation.

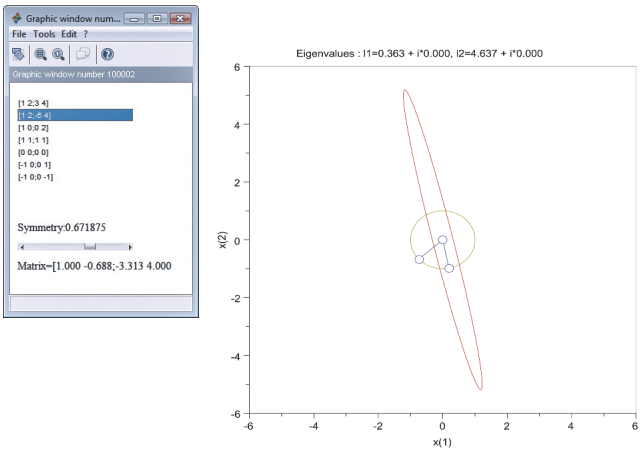

Figure 3: A GUI to display the eigenvalues and eigenvectors of a 2-by-2 matrix.

Focus on Differential Equations

- 9 solvers for ODEs and integration : ode, bvode, dassl, etc.
- Computes the solution y(t) of dy(t)=dt = f(y) with x real vector.
- Includes a collection of algorithms:
  - Stiff problems: Backward Differentiation Formula (BDF) from Odepack,
  - Non-stiff problems: Adams method from Odepack,
  - Adaptive Runge-Kutta of order 4 (RK4),
  - Shampine and Watts program based on Fehlberg's Runge-Kutta pair of order 4 and 5,
  - Root finding from Odepack,
  - Discrete time simulation.

- This module can manage the following type of right hand side of the first order differential equation:
  - Macros (functions), with, if required, optional arguments,
  - Fortran or C dynamically linked functions.

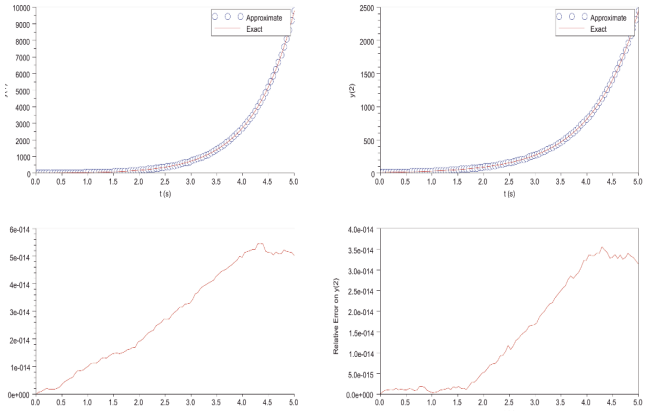

Figure 4: Comparison between the exact and approximated solutions of the linear Ordinary Differential Equation dy/dt=Ay(t).

A set of robust libraries

- Quadpack: a package for the numerical computation of definite one-dimensional integrals.

http://www.netlib.org/quadpack

- Odepack: a collection of Fortran solvers for the initial value problem for ordinary differential equation systems.

http://www.netlib.org/odepack

- Modulopt: a collection of optimization solvers.

http://www-rocq.inria.fr/~gilbert/modulopt/

- Minpack: a Fortran 77 code for solving nonlinear equations and nonlinear least squares problems.

http://www.netlib.org/minpack/

- Dcdflib: a Fortran 77 code for cumulative distribution functions, inverses, and parameters.

http://www.netlib.org/random/

- Many macros: e.g. derivative.

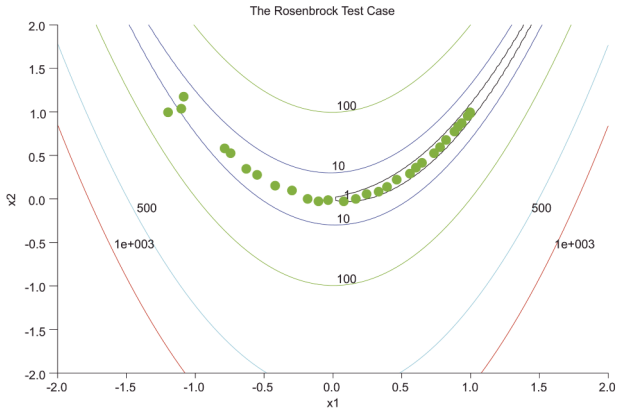

Figure 5: Optimization of the Rosenbrock function by the optim function.

Why you should use Scilab for classical numerical methods

- Accuracy: all algorithms have be designed by experts for floating point computations.
- Flexibility: we can drive either Scilab-defined or external codes in C or Fortran - Plug your own simulation code with dynamic linking.
- Robustness: most algorithms are used for decades.

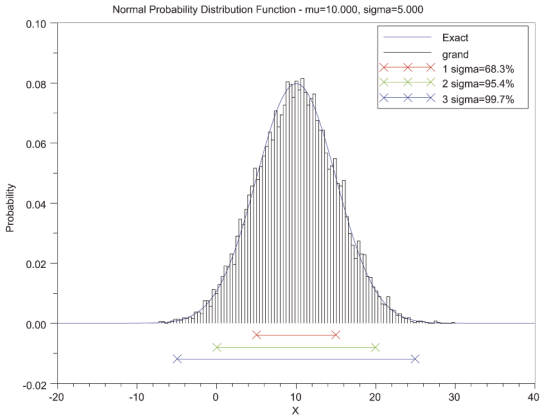

Figure 6: The normal distribution function and the grand function.

Advanced Numerical Features

- Control: classical and robust control,
- Signal Processing,
- Sparse Matrices: Sparse Gaussian Elimitation, Sparse Cholesky Decomposition, Connection to the UMFPACK and TAUCS libraries,
- Many more numerical tools via ATOMS.