Optimization with Scilab, present and future.

Serge Steer
INRIA Rocquencourt BP105
78153 Le Chesnay Cedex, France
Email: Serge.Steer@inria.fr,
Michael Baudin
INRIA Rocquencourt BP105
78153 Le Chesnay Cedex, France
Email: Michael.Baudin@Scilab.org

July 16, 2009

1 Extended abstract

Minimize cost, weight, consumption, maximize the performance, ... are increasingly common queries in the work of engineers.

Often there is no "one size fits all" design which works well in all cases, so engineers make trade-offs to optimize the attributes of greatest interest.

Most optimization problems can be mathematically stated as follow:

$$\begin{cases} \underset{x \in I}{\text{minimize}} f(x) \\ \text{with} \\ f : I \subset \mathbb{R}^n \to \mathbb{R} \\ \text{Subject to} \\ g_i(x) \leq 0,\ g_i : I \to \mathbb{R},\ \ i = 1 : m_i \\ h_j(x) = 0,\ h_j : I \to \mathbb{R},\ j = 1 : m_j \end{cases}$$

Where "minimize" may mean "find a value of x near a local or a global minimum ".

The available optimization's methods depend strongly on the criterion f (linear, quadratic, non linear, min max), on the constraints g_i h_j (bounds, linear, quadratic or general), and on the size of the problem, but also the on the expected quality of approximation of the solution. In this context, it is essential that the engineers can have, on their computing platforms, a wide range of optimization methods and tools that are easily usable and interchangeable. The paper will present the functions available in the Scilab distribution as well as the contributed modules. These functions cover

the linear problems (lp_solve, linpro) the quadradic problems (qpsolve, qld, quapro), non linear least square problems (lsqrsolve, leastsq), non linear general problems (optim, conmin, fsqp, newuoa), semidefinite programming (semidef, lmisolver) optimization in graphs and networks (shortest path, minimum weight tree, knapsak, salesman, minimum linear and quadratic flows)...

It is very difficult to verify the solution proposed by an optimization algorithm. Also, it is important that the algorithms proposed are tested on large test sets. A large bunch is freely available in the net but the tests are described using various languages (AMPL, CUTEr, MPS, ...). The paper will present the Scilab tools available to adress these test sets, as well as some bench results obtains with Scilab optimization tools run on these tests.

The paper will present the future work that we intend to do in order to extend and improve Scilab optimization related tools. This action will mainly consist in adding new solvers like GLPK (GNU Linear Programming Kit) which solves LP and MIP problems, SeDuMi for linear, second order, semidefinite or mixed problems or some global optimization tools (Monte-Carlo, Simulated annealing, ...).

We will also present the object-oriented approach that could be used in Scilab environment to enter the optimization problems in a simple and flexible enough way allowing their resolution by different solvers.