1 Package nelder_mead

1.1 Introduction to package nelder_mead

Package `nelder_mead` implements the Nelder-Mead minimization algorithm [1], also known as the polytope or amoeba method.

The Nelder-Mead algorithm is a derivative-free minimization algorithm; only evaluations of the objective function are required.

`nelder_mead` implements the "grid-restrained" Nelder-Mead algorithm published by A. Brmen et al. [2], and implemented in Common Lisp by Mario S. Mommer. Thanks to Andrej Vodopivec for the Maxima interface to the Common Lisp code.

References

[1] J.A. Nelder and R. Mead, "A simplex method for function minimization," The Computer Journal, vol. 7, pp. 308-313, 1965.

[2] A. Brmen, J. Puhan and T. Tuma, "Grid Restrained Nelder-Mead Algorithm", Computational Optimization and Applications, vol. 34, no. 3, pp. 359 - 375, 2006.

1.2 Definitions for package nelder_mead

`nelder_mead` (*obj*, *vars*, *init*) [Function]

Returns an approximate minimum of the objective function *obj*, as a function of the variables *vars*, starting at the initial point *init*.

The objective function may be discontinuous, but if it is continuous, the algorithm ("grid-restrained" Nelder-Mead) is provably convergent. The objective function need not be differentiable; derivatives are not computed, not even approximately.

Examples:

```
(%i1) load ("nelder_mead") $
(%i2) nelder_mead (if x<0 then -x else x^2, [x], [4]);
(%o2)                [x = 9.536387892694628E-11]
(%i3) f(x) := if x < 0 then -x else x^2 $
(%i4) nelder_mead (f, [x], [4]);
(%o4)                [x = 9.536387892694628E-11]
(%i5) nelder_mead (x^4 + y^4 - 2*x*y - 4*x - 3*y, [x, y], [2, 2]);
(%o5)    [x = 1.157212489168102, y = 1.099342680267472]
```

Appendix A Function and variable index

(Index is nonexistent)