

NAME

CUTEST_cidh – CUTEst tool to evaluate the Hessian of a problem function.

SYNOPSIS

CALL CUTEST_cidh(status, n, X, iprob, lh1, H_val)

DESCRIPTION

The CUTEST_cidh subroutine evaluates the Hessian matrix of either the objective function or a constraint function for the problem decoded from a SIF file by the script *sifdecode* at the point X, and possibly its gradient. The matrix is stored as a dense matrix.

The problem under consideration is to minimize or maximize an objective function $f(x)$ over all $x \in R^n$ subject to general equations $c_i(x) = 0$, ($i \in 1, \dots, m_E$), general inequalities $c_i^l(x) \leq c_i(x) \leq c_i^u(x)$, ($i \in m_E + 1, \dots, m$), and simple bounds $x^l \leq x \leq x^u$. The objective function is group-partially separable and all constraint functions are partially separable.

ARGUMENTS

The arguments of CUTEST_cidh are as follows

status [out] - integer

the output status: 0 for a successful call, 1 for an array allocation/deallocation error, 2 for an array bound error, 3 for an evaluation error,

n [in] - integer

the number of variables for the problem,

X [in] - real/double precision

an array which gives the current estimate of the solution of the problem,

iprob [in] - integer

the number of the problem function to be considered. If iprob = 0, the Hessian of the objective function will be evaluated, while if iprob = i > 0, that of the i-th constraint will be evaluated,

lh1 [in] - integer

the actual declared size of the leading dimension of H_val (with lh1 no smaller than n),

H_val [out] - real/double precision

a two-dimensional array which gives the value of the required Hessian matrix.

AUTHORS

I. Bongartz, A.R. Conn, N.I.M. Gould, D. Orban and Ph.L. Toint

SEE ALSO

CUTEr (and SifDec): A Constrained and Unconstrained Testing Environment, revisited,

N.I.M. Gould, D. Orban and Ph.L. Toint,

ACM TOMS, **29**:4, pp.373-394, 2003.

CUTE: Constrained and Unconstrained Testing Environment, I. Bongartz, A.R. Conn, N.I.M. Gould and Ph.L. Toint, TOMS, **21**:1, pp.123-160, 1995.

sifdecode(1).

NAME

CUTEST_cidh – CUTEst tool to evaluate the Hessian of a problem function.

SYNOPSIS

CALL CUTEST_cidh(status, n, X, iprob, lh1, H_val)

DESCRIPTION

The CUTEST_cidh subroutine evaluates the Hessian matrix of either the objective function or a constraint function for the problem decoded from a SIF file by the script *sifdecode* at the point X, and possibly its gradient. The matrix is stored as a dense matrix.

The problem under consideration is to minimize or maximize an objective function $f(x)$ over all $x \in R^n$ subject to general equations $c_i(x) = 0$, ($i \in 1, \dots, m_E$), general inequalities $c_i^l(x) \leq c_i(x) \leq c_i^u(x)$, ($i \in m_E + 1, \dots, m$), and simple bounds $x^l \leq x \leq x^u$. The objective function is group-partially separable and all constraint functions are partially separable.

ARGUMENTS

The arguments of CUTEST_cidh are as follows

status [out] - integer

the output status: 0 for a successful call, 1 for an array allocation/deallocation error, 2 for an array bound error, 3 for an evaluation error,

n [in] - integer

the number of variables for the problem,

X [in] - real/double precision

an array which gives the current estimate of the solution of the problem,

iprob [in] - integer

the number of the problem function to be considered. If iprob = 0, the Hessian of the objective function will be evaluated, while if iprob = i > 0, that of the i-th constraint will be evaluated,

lh1 [in] - integer

the actual declared size of the leading dimension of H_val (with lh1 no smaller than n),

H_val [out] - real/double precision

a two-dimensional array which gives the value of the required Hessian matrix.

AUTHORS

I. Bongartz, A.R. Conn, N.I.M. Gould, D. Orban and Ph.L. Toint

SEE ALSO

CUTEr (and SifDec): A Constrained and Unconstrained Testing Environment, revisited,

N.I.M. Gould, D. Orban and Ph.L. Toint,

ACM TOMS, **29**:4, pp.373-394, 2003.

CUTE: Constrained and Unconstrained Testing Environment, I. Bongartz, A.R. Conn, N.I.M. Gould and Ph.L. Toint, TOMS, **21**:1, pp.123-160, 1995.

sifdecode(1).