**NAME**

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

**SYNOPSIS**

CALL CUTEST_cidh_threaded( status, n, X, iprob, lh1, H_val, thread )

For real rather than double precision arguments, instead

CALL CUTEST_cidh_threaded_s( ... )

and for quadruple precision arguments, when available,

CALL CUTEST_cidh_threaded_q( ... )

**DESCRIPTION**

The CUTEST_cidh_threaded 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 *sifdecoder* 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 $c_i(x) = 0$, $(i \in 1, \ldots, m_E)$, general inequalities $c_i^l \le c_i(x) \le c_i^{u,}$ $(i \in m_E + 1, \ldots, m)$, and simple bounds $x^l \le x \le x^{u.}$ The objective function is group-partially separable and all constraint functions are partially separable.

**ARGUMENTS**

The arguments of CUTEST_cidh_threaded are as follows

**status** [out] - integer

the outputr status: 0 for a succesful call, 1 for an array allocation/deallocation error, 2 for an array bound error, 3 for an evaluation error, 4 for an out-of-range thread,

**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,

**irpob** [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,

**thread** [in] - integer

thread chosen for the evaluation; threads are numbered from 1 to the value threads set when calling CUTEST_csetup_threaded.

**AUTHORS**

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

**SEE ALSO**

*CUTEst: a Constrained and Unconstrained Testing Environment with safe threads*,
  N.I.M. Gould, D. Orban and Ph.L. Toint,
  Computational Optimization and Applications **60**:3, pp.545-557, 2014.

*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,
  ACM TOMS, **21**:1, pp.123-160, 1995.

sifdecoder(1).