NAME

CUTEST_usetup_threaded − CUTEst tool to set up the data structures for unconstrained or bound-constrained minimization.

SYNOPSIS

CALL CUTEST_usetup_threaded(status, input, out, threads, IO_BUFFERS, n, X, X_l, X_u)

For real rather than double precision arguments, instead

CALL CUTEST_usetup_threaded_s(...)

and for quadruple precision arguments, when available,

CALL CUTEST_usetup_threaded_q(...)

DESCRIPTION

The CUTEST_usetup_threaded subroutine sets up the correct data structures for subsequent threaded computations in the case where the only possible constraints are bound constraints.

The problem under consideration is to minimize or maximize an objective function $f(x)$ over all $x \in R^n$ subject to the simple bounds $x^l \leq x \leq x^u.$ The objective function is group-partially separable.

ARGUMENTS

The arguments of CUTEST_usetup_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,

input [in] - integer
the unit number for the decoded data; the unit from which OUTSDIF.d is read,

out [in] - integer
the unit number for any error messages,

threads [in] - integer
the total number of independent evaluation threads that are required,

IO_BUFFERS [in] - integer
an array of different unit numbers, one entry for each thread, for any internal input/output,

n [out] - integer
on input, the declared dimesions of X, X_l and X_u (see argument n in CUTEST_udimen). On output, the number of variables for the problem,

X [out] - real/double precision
an array that gives the initial estimate of the solution of the problem,

X_l [out] - real/double precision
an array that gives lower bounds on the variables,

X_u [out] - real/double precision
an array that gives upper bounds on the variables.

APPLICATION USAGE

A call to CUTEST_usetup_threaded must precede calls to other threaded evaluation tools, except CUTEST_udimen, for unconstrained and bound constrained problems.

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.

cutest_csetup_threaded(3M), cutest_udimen(3M), sifdecoder(1).