

**NAME**

CUTEST\_ccfsg\_threaded – CUTEst tool to evaluate constraint functions values and possibly their gradients in sparse format.

**SYNOPSIS**

CALL CUTEST\_ccfsg\_threaded( status, n, m, X, C, nnzj, lj, J\_val, J\_var, J\_fun, grad, thread )

**DESCRIPTION**

The CUTEST\_ccfsg\_threaded subroutine evaluates the values of the constraint functions of the problem decoded from a SIF file by the script *sifdecoder* at the point X, and possibly their gradients in the constrained minimization case. The gradients are stored in sparse format.

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  $c_i^l \leq c_i(x) \leq c_i^u$  ( $i \in m_E + 1, \dots, m$ ), 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\_ccfsg\_threaded 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, 4 for an out-of-range thread,

**n** [in] - integer

the number of variables for the problem,

**m** [in] - integer

the total number of general constraints,

**X** [in] - real/double precision

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

**C** [out] - real/double precision

an array which gives the values of the general constraint functions evaluated at X. The i-th component of C will contain the value of  $c_i(x)$ .

**nnzj** [out] - integer

the number of nonzeros in J\_val,

**lj** [in] - integer

the actual declared dimensions of J\_val, J\_var and J\_fun,

**J\_val** [out] - real/double precision

an array which gives the values of the nonzeros of the general constraint functions evaluated at X. The i-th entry of J\_val gives the value of the derivative with respect to variable J\_var(i) of constraint function J\_fun(i),

**J\_var** [out] - integer

an array whose i-th component is the index of the variable with respect to which J\_val(i) is the derivative,

**J\_fun** [out] - integer

an array whose i-th component is the index of the problem function of which J\_val(i) is the derivative,

**grad** [in] - logical

a logical variable which should be set .TRUE. if the gradient of the constraint functions are required and .FALSE. otherwise,

**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)