NAME

CUTEST_csjp – CUTEst tool to evaluate the sparsity pattern of the Jacobian of constraints gradients.

SYNOPSIS

CALL CUTEST_csjp(status, nnzj, lj, J_var, J_con)

DESCRIPTION

The CUTEST_csjp subroutine evaluates the sparsity pattern used when storing the Jacobian matrix of gradients of the general constraints corresponding to the problem decoded from a SIF file by the script *sifdecoder*.

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_csjp 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,

nnzj [out] - integer
the number of nonzeros in J_var and J_con,

lj [in] - integer
the actual declared dimensions of J_var and J_con,

J_var [out] - integer
an array whose i-th component is the index of the variable with respect to which the derivative is taken,

J_con [out] - integer
an array whose i-th component is the index of the constraint function whose derivative is taken.

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_csgr(3M), sifdecoder(1).